APPLE II C

VOTRE MICRO-ORDINATEUR

collection
micro
monde





APPLE //c

VOTRE MICRO-ORDINATEUR

Eric Duceau et Christophe Doë



dirigée par Serge Pouts-Lajus



Dans la même collection

La micro en 100 questions
Bruno de Latour

Dictionnaire micro-informatique
Éric Duceau et Christophe Doë

Choisir son micro-ordinateur
Yvon Dargery

Portatifs, les nouveaux micro-ordinateurs
Pierre Raguenes et Gérard Sitbon

Logo, votre langage de programmation
André Myx et Pierre Subtil

BASIC, votre langage de programmation
Serge Pouts-Lajus

MO5, votre micro-ordinateur
Serge Pouts-Lajus

MO5, TO7, TO7-70, vos programmes
Pierre Champeaux et Serge Pouts-Lajus

Macintosh, votre micro-ordinateur
Jean-Baptiste Touchard

Minitel, votre guide pratique
Jacques David

Alice, Alice 90, votre micro-ordinateur
Jean Delcourt

Commodore 64, votre micro-ordinateur
Daniel Isidore

Oric-Atmos, votre micro-ordinateur
Michel Bussac

Oric-Atmos, vos programmes
Michel Bussac et Gill Espèche

ZX-Spectrum, votre micro-ordinateur
Serge Pouts-Lajus

Couverture : Studio Bercy
Photos reproduites avec l'autorisation d'Apple Seedrin
et de Golden.

Ce volume porte la référence
ISBN 2-7124-1517-5

CEDIC, 32, boulevard Saint-Germain, 75005 - PARIS

SOMMAIRE

Présentation, installation 5

Votre Apple //c 6
Le microprocesseur 15
La mémoire 23
Les disquettes 25

Le système d'exploitation 29

Qu'est-ce qu'un système d'exploitation ? 32
PRODOS 36
Les utilitaires-système 41

Les progiciels 51

Les traitements de texte 56
Les tableurs 65
Les gestionnaires de fichiers 71
Les logiciels intégrés 77
Un logiciel à part : THINKTANK 84

Les langages 89

PASCAL 89
BASIC 91
LOGO 93

Les périphériques 99

Les transmissions 100
Les moniteurs 104
La souris 111
Les imprimantes 114

Conclusion 125

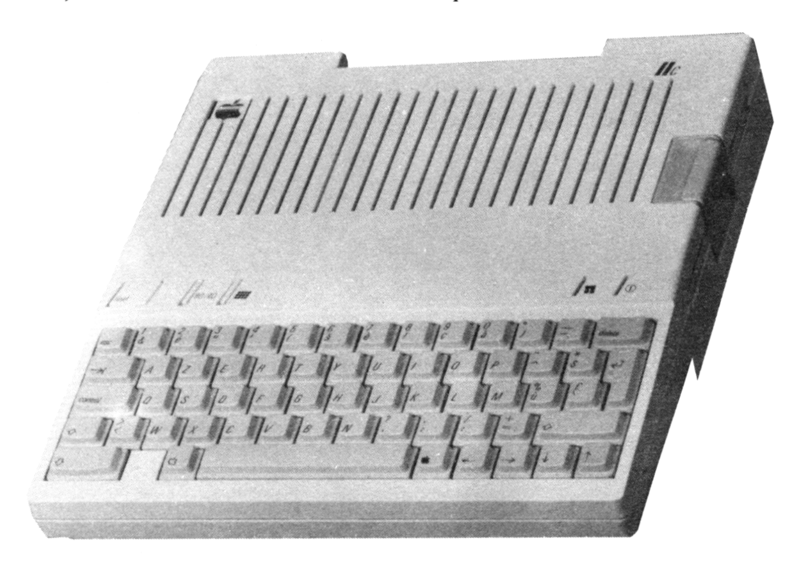
PRESENTATION, INSTALLATION

Apple //c est l'héritier d'Apple //e, figure légendaire de la micro-informatique. Les cousins Lisa et Macintosh sont en passe de devenir des vedettes de la technologie de pointe. L'Apple //c se situe au carrefour de la tradition Apple : l'expérience du //e ajoutée aux innovations du Macintosh (la souris et un puissant système d'exploitation). Les ambitions de l'Apple //c sont grandes.

C'est pourquoi l'aspect micro-ordinateur professionnel est particulièrement mis en avant. On retrouve cette orientation dans les périphériques disponibles. Les écrans de visualisation prévus sont petits et performants ; un écran à cristaux liquides est maintenant disponible, accentuant encore les possibilités de portabilité du //c. De même, la nouvelle imprimante matricielle prévue pour l'Apple //c est plus performante que ses rivales dans la même gamme de prix.

VOTRE APPLE //c

• La machine se présente sous la forme d'un bloc comprenant l'unité centrale, le clavier et un lecteur de disquettes.



• Elle est livrée avec un certain nombre de fils de connexion dont la prise Péritel qui permet de la brancher sur un téléviseur.

• **Remarque :** un système de visualisation est nécessaire et le choix doit s'opérer entre un téléviseur et un moniteur. Les différences entre ces deux options vous seront présentées dans le chapitre "Périphériques".

• L'alimentation de l'Apple //c est séparée du boîtier central. Un petit transformateur est fourni avec le micro-ordinateur. La nouvelle technologie développée pour la construction du //c permet une consommation très faible et autorise un branchement à des sources d'électricité portatives (batterie de 12 volts par exemple). Les amateurs de déplacements "sur le terrain" apprécieront.

• Le bloc unité-centrale contient le microprocesseur et la mémoire de l'ordinateur. Quelques chiffres pour situer l'Apple //c :
— Microprocesseur 8 bits 65C02
— Mémoire centrale de 128 K

Ces performances seront expliquées plus loin ; sachez seulement que ce micro-ordinateur dispose d'une capacité mémoire supérieure à la moyenne de ses rivaux classiques et que son microprocesseur est réputé pour sa rapidité.

• Le clavier ressemble en tout point à celui d'une machine à écrire, c'est-à-dire que l'on y trouve tous les caractères de la langue française. Quelques touches spéciales ont des fonctions que vous découvrirez à l'utilisation :

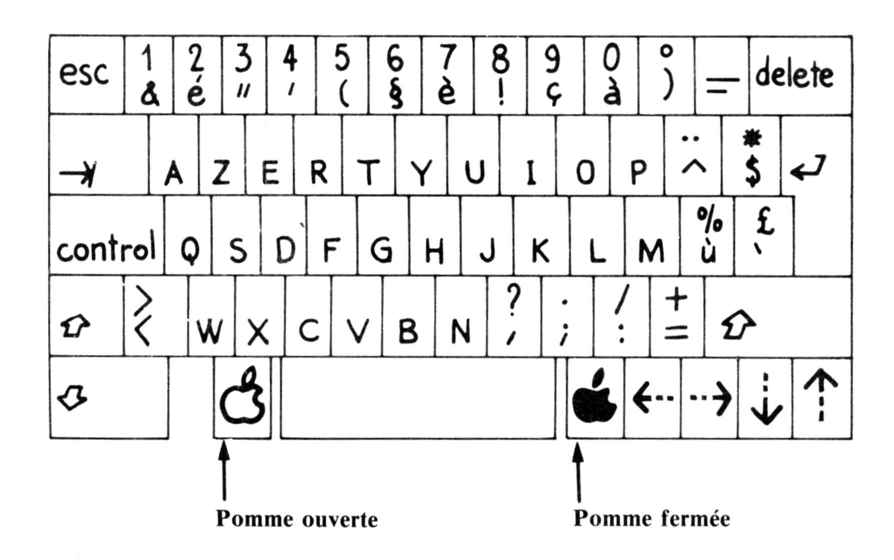
— la touche "RETURN" permet de signaler au micro-ordinateur qu'il doit prendre en compte ce qui vient d'être tapé ;

— les touches "CONTROL" et "POMME" sont utilisées en combinaison avec d'autres :

1) "POMME OUVERTE" + "CONTROL" + "RESET" pressées puis lachées simultanément produisent le chargement d'une disquette du système d'exploitation (la disquette "utilitaires-système" par exemple).

2) "CONTROL" + "RESET" vous donnent accès au BASIC.

Toutes ces manipulations demandent à être apprises et vous sont expliquées dans le manuel d'utilisation.



Pomme ouverte

Pomme fermée

• Votre Apple //c dispose d'un lecteur de disquettes intégré. Pour un micro-ordinateur, les disquettes représentent le procédé de stockage privilégié (mémoire de masse) : la puissance de la machine dépend des utilisations possibles de cette mémoire externe. En particulier, il faut pouvoir échanger rapidement et précisément des informations entre la RAM (mémoire interne) et la disquette.

Le système d'exploitation de disquette (en abrégé SED, ou DOS pour Disk Operating System) est un ensemble de programmes dont la fonction est de s'occuper des échanges, du stockage et de la restitution des informations. Il est livré avec le micro-ordinateur.

• Quelques options disponibles directement :

— Une première touche vous permet de choisir un affichage sur 40 colonnes ou 80 colonnes à l'écran. Mais attention : certaines configurations ou certaines options ont priorité sur votre choix.

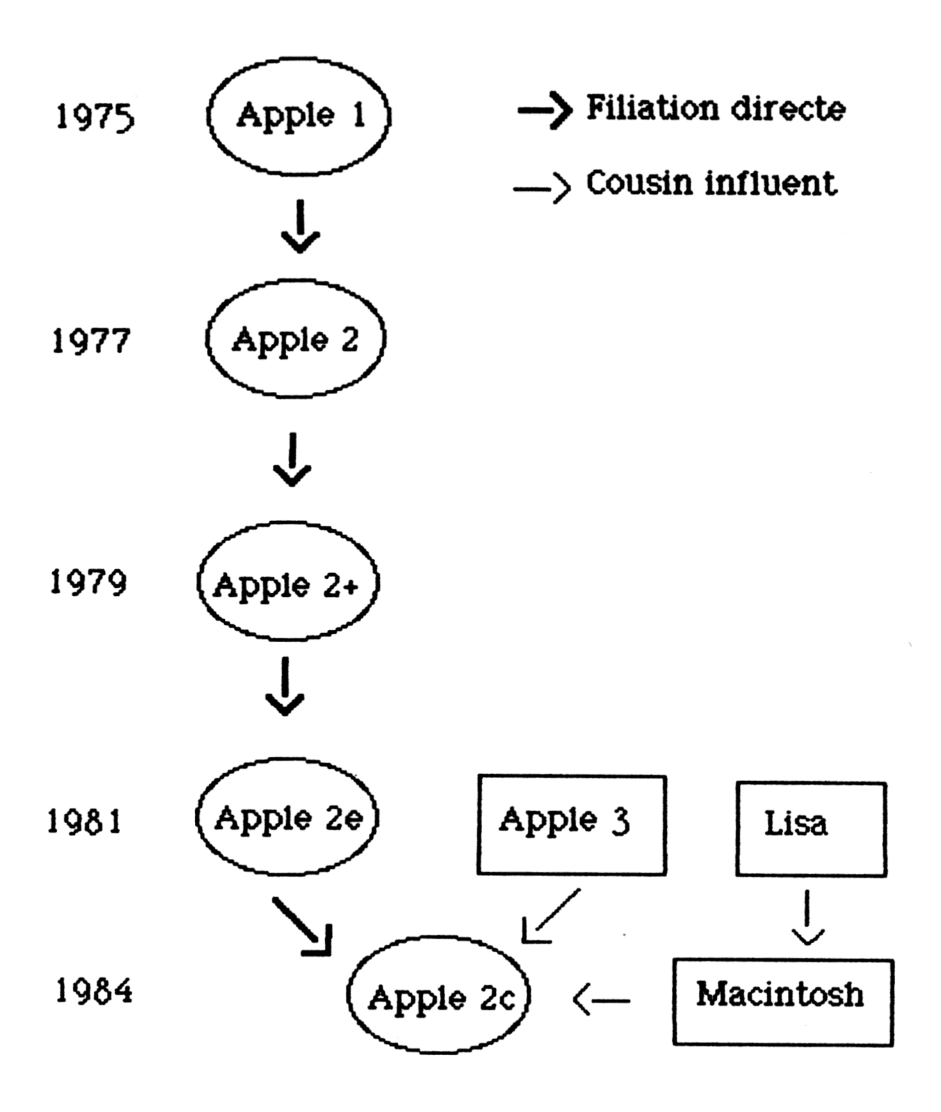
Exemple : 1) Si vous utilisez un téléviseur, l'option 80 colonnes est inopérante.
2) Si vous chargez certains progiciels (Applewriter entre autres), l'option 40 colonnes n'est pas accessible sur un moniteur.



— Le clavier proposé est français (AZERTY) mais les amateurs de clavier américain (QWERTY) peuvent enfoncer la touche prévue à cet effet. Il faudra alors penser à rétablir les correspondances entre £ et # par exemple et retrouver les symboles spéciaux [,], etc.

— En cas d'erreur, le micro-ordinateur vous prévient simultanément à l'écran et en émettant un bip sonore. Le son se règle à l'aide d'un potentiomètre.

APPLE //c DANS LA GAMME APPLE



Arbre généalogique de la famille Apple

Comme on le voit sur cet arbre généalogique, l'Apple //c se place dans la droite ligne de l'Apple //e. Il a subi des influences marquantes de la part des "cousins" Macintosh et Lisa mais les performances du //c dépendent beaucoup de ses ancêtres directs.

Par exemple, pour pouvoir utiliser les mêmes disquettes qu'avec le //e, le //c dispose du même lecteur. Celui-ci a fait ses preuves mais les technologies récentes montrent que ses performances sont quelconques : stocker 140 Ko sur une face de disquette est à la portée de tout bon micro-ordinateur. Le désir de compatibilité avec les précédentes versions a prévalu.

En conséquence, les systèmes d'exploitation ont subi la même lente évolution. Leur implantation dans la mémoire de l'ordinateur s'explique également par l'histoire. Les premiers Apple ne disposaient pas de lecteurs de disquettes et le système d'exploitation était stocké en mémoire morte. C'est pourquoi on trouve toujours le même schéma de séparation du système d'exploitation entre la disquette des utilitaires et la mémoire morte.

Les cousins plus puissants comme Lisa ont apporté un nouveau système d'exploitation capable de bien meilleures performances. En particulier, on peut imaginer gérer un disque dur avec l'Apple //c, ce qu'il était impossible de faire avec les autres Apple //.

POURQUOI c ?

Le dernier-né des Apple // est un compact : voilà l'explication du petit "c". Par compact, on entend surtout portable, c'est-à-dire que l'utilisateur peut le transporter avec lui facilement. Une poignée est d'ailleurs prévue à cet effet.

Remarque : Cette poignée a aussi une fonction vitale que le manuel vous rappellera régulièrement. Elle permet de surélever l'arrière de l'Apple //c pour faciliter l'aération.

Les chiffres
— poids : 3,4 kg
— encombrement : 30 × 28 × 6,5 cm

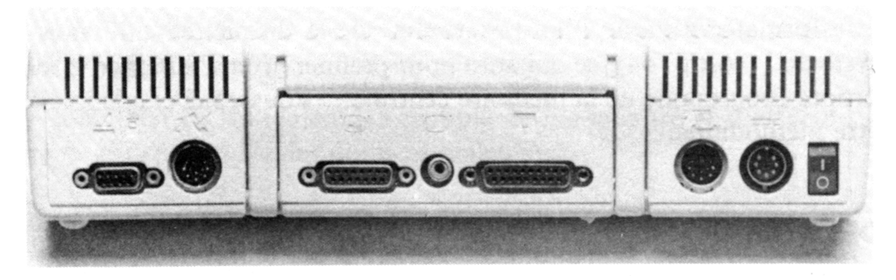
Pour compléter l'aspect portable, il faut penser à l'écran à cristaux liquides dont la taille et la consommation s'inscrivent parfaitement dans ce contexte.

CONNEXIONS

Par opposition aux autres Apple //, le dernier-né est un système fermé, c'est-à-dire qu'il n'est pas question de le démonter pour lui ajouter des cartes d'extension et "bidouiller" un quelconque branchement. Les concepteurs du //c ont donc "tout prévu" et l'arrière du micro-ordinateur est pourvu de toutes les sorties (ports) nécessaires.

On repère la sortie utile grâce aux icônes. Depuis quelque temps, la mode est aux icônes, petits dessins symbolisant une option ou un matériel, dont nous reparlerons abondamment à propos des progiciels. On peut donc facilement connecter plusieurs périphériques :

- Un moniteur ou une télé (2 sorties distinctes)
- Un lecteur de disquettes supplémentaire
- Une imprimante et un modem
- Une souris et des manettes de jeu (même sortie)



Les fiches ne sont pas interchangeables (excepté modem et imprimante). Vous n'avez donc aucun risque de brancher le lecteur externe sur la prise Péritel !

Déterminer l'environnement de votre micro-ordinateur, c'est-à-dire la configuration en périphériques et logiciels qui vous convient, doit être votre premier souci.

Pour toutes les applications professionnelles, une sérieuse gamme de progiciels existe déjà. Les disquettes de présentation de l'Apple //c fournies lors de la livraison vous présentent les applications les plus courantes : le traitement de texte, la gestion de fichiers et le travail sur feuille de calcul électronique. Nous détaillerons plus loin les possibilités de tels logiciels.

BASIC ?

N'oublions pas que l'utilisateur est aussi un programmeur potentiel. Pour obtenir BASIC dès la mise sous tension, il suffit d'enfoncer simultanément les touches "CONTROL" et "RESET". Le curseur apparaîtra alors et vous disposerez du classique BASIC Applesoft. Une documentation complète est livrée avec le micro-ordinateur et elle vous guidera pas à pas dans le BASIC du //c.

On remarquera que le BASIC de l'Apple //c est un "vieux BASIC". C'est la compatibilité avec l'ancêtre Apple //e qui l'exige. L'éditeur de texte ne vous permet pas facilement de vous déplacer sur tout l'écran. L'apprentissage est un peu plus long que pour certaines autres machines mais à la longue, on se fait aux manipulations avec le mode "Escape".

Mais ne vous lancez pas hâtivement dans le développement d'un gros programme. Vous devez préalablement songer à la sauvegarde sur une disquette des fruits de vos efforts. Cette disquette doit d'abord être formatée à l'aide d'un programme de la disquette "utilitaires-système" (voir p. 44), ce qui aura pour premier effet d'éliminer votre propre programme de la mémoire centrale... Vous voici condamné à être méthodique.

DES NOMS !

• Steve Jobs et Steve Wozniack : ils sont les fondateurs d'Apple. A 20 ans, ils bricolaient dans un garage le premier d'une longue série. Quoique milliardaires, ils n'ont rien perdu de leur dynamisme.

• Hartmut Esslinger : c'est le designer de l'Apple //c. On lui devait le fameux Walkman Sony, premier de sa catégorie. On raconte que Jobs tyrannisa Esslinger jusqu'à ce que la coque du //c fût comme elle devait être.

• Snow White : ainsi a failli s'appeler le //c. Au dernier moment, ils n'ont pas osé. Faut-il le regretter ?

• Apple Fog : c'est la couleur de l'Apple //c. Reposante, rassurante et un petit air sérieux qui plaira aux gens sérieux.

QU'EST-CE QU'UN MICRO-ORDINATEUR ?

La fonction essentielle d'un micro-ordinateur est le traitement de l'information, c'est-à-dire l'enregistrement, l'archivage, les calculs, les tris, les contrôles, etc.

Pour cela, l'ordinateur doit pouvoir disposer des fonctions suivantes : saisie de l'information, mémorisation, calculs, restitution.

Un micro-ordinateur est donc composé des unités fonctionnelles suivantes :

— des périphériques dits d'entrée/sortie tels que les claviers, les écrans, les imprimantes, qui lui permettent de communiquer avec le monde extérieur (saisie, restitution) ;

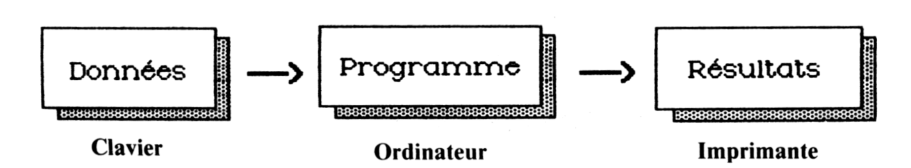
— de la mémoire pour pouvoir stocker les informations qu'il traite (mémorisation) ;

— une unité de traitement susceptible de reconnaître et d'effectuer des modifications sur les données (calculs).

Le traitement de l'information s'effectue à l'aide d'instructions élémentaires identifiables et exécutables. Cette suite d'instructions constitue ce que l'on appelle un *programme*.

Notons que les programmes doivent être également mémorisés et traités. En ce sens, ils ne se distinguent pas réellement des données qu'il manipulent.

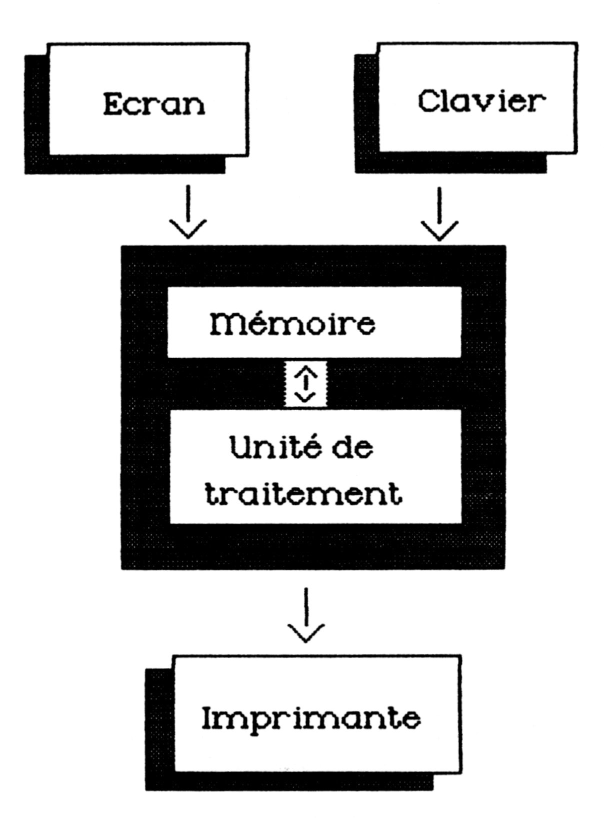
Exemple :



Différents langages sont utilisés pour que l'homme puisse communiquer (écrire des programmes) avec la machine.
Ils se distinguent par leurs analogies avec le langage naturel, leur complexité, leur orientation vers différents types d'applications. Ils se caractérisent par leur alphabet, leur orthographe et leur grammaire.

En fait, ces langages ont été créés par l'homme et pour l'homme mais l'ordinateur étant une machine, le langage est essentiellement fondé sur le passage de signaux électriques.

Il existe donc, au sein de la machine, des programmes qui convertissent les instructions du langage en une suite de signaux électriques "compréhensibles". Ces signaux transitent entre les diverses unités de la machine et peuvent être stockés en mémoire. Il faut donc voir un micro-ordinateur comme une "machine à communiquer de l'information, à la stocker, à la traiter".



Synoptique d'un micro-ordinateur

LE MICROPROCESSEUR

Un microprocesseur se présente sous l'aspect d'une puce de silicium supportant un ensemble de circuits intégrés, enfermée dans un boîtier en plastique muni de connexions métalliques.

La caractéristique essentielle d'un microprocesseur est son aspect programmable : il se distingue en cela d'un automate qui ne peut effectuer qu'un nombre restreint de tâches pour lesquelles il a été conçu ; un microprocesseur est capable d'aller rechercher le programme qu'il doit effectuer en mémoire centrale,puis de l'exécuter.

On peut, pour fixer les idées, estimer à plus de dix mille le nombre de transistors enfermés dans un microprocesseur.

Dans un micro-ordinateur, le microprocesseur est essentiellement chargé d'effectuer tous les calculs arithmétiques et logiques, nécessaires au déroulement des programmes. Mais il n'est pas seulement capable de calculs. Il organise et synchronise dans le temps l'exécution des diverses instructions de ces programmes.

Les micro-ordinateurs sont susceptibles de traiter un nombre restreint d'instructions, en général une centaine, appelées ''jeu d'instructions'', variables d'un microprocesseur à l'autre.

On peut regrouper ces instructions en plusieurs types :

• les mouvements de données entre la mémoire centrale et la mémoire personnelle du microprocesseur.
Elles sont du type : ''aller écrire à l'adresse xxxx le contenu du registre A'' ou ''charger le registre X avec le contenu de la case mémoire d'adresse xxxx ;

• les opérations arithmétiques telles que l'addition, la soustraction, l'incrément de compteur.
Ces opérations ne peuvent agir que sur le contenu d'un registre ;

• les opérations logiques telles que ET, OU, OU exclusif ;

• les instructions de branchements conditionnels et inconditionnels qui permettent d'interrompre le déroulement séquentiel d'un programme.

En ce qui concerne les branchements conditionnels, la condition porte généralement sur des bits appartenant à un registre particulier appelé registre d'indicateur.

• les opérations concernant un ou plusieurs bits appartenant à un octet. Il existe en effet des instructions qui permettent d'effectuer des décalages à droite ou à gauche, bit par bit.

• les instructions de branchement et d'appel de sous-programmes ;

• la gestion des interruptions, provoquées par exemple par l'entrée d'un caractère au clavier, ou par la lecture de données inscrites sur une disquette.

Ces instructions sont naturellement codées et traitées en binaire. Cependant, il existe pour chacune d'entre elles une équivalence mnémonique qui facilite la programmation du microprocesseur.

Exemples de mnémoniques :

LDA pour "load accumulator from memory" qui affecte une valeur à un accumulateur.

STA pour "store accumulator" qui transfère une donnée d'un accumulateur vers une adresse mémoire.

JMP pour "jump", saut inconditionnel à une instruction.

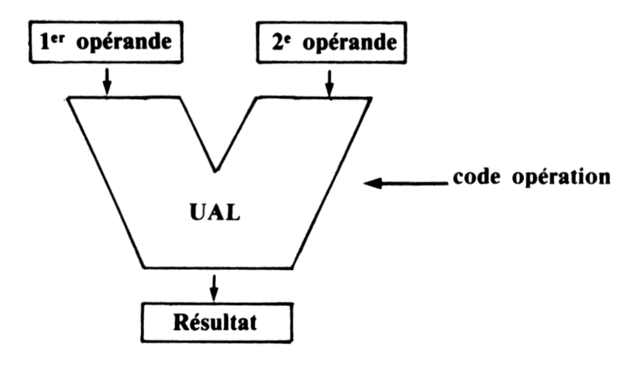
Un programme nommé assembleur effectue la traduction de ces mnémoniques en code binaire et des adresses symboliques en adresses réellement implantées en mémoire.

Notons que l'assembleur n'est pas résident dans l'Apple //c : on ne peut donc programmer directement dans ce langage. Il est nécessaire d'acheter dans le commerce une disquette particulière dite "disquette assembleur".

L'unité arithmétique et logique (UAL)

C'est l'ensemble des circuits chargés d'effectuer les opérations arithmétiques, addition, multiplication et les opérations logiques, ET, OU entre les variables apparaissant dans le programme à exécuter.

On peut en faire la représentation schématique suivante :



UAL : Structure à deux opérandes

Cette unité ne sait traiter qu'une seule opération en même temps, comprenant au plus deux opérandes. Elle reçoit donc en entrée le type d'opération à effectuer et les variables sur lesquelles portera cette opération. A la sortie, elle dépose le résultat de l'opération dans une mémoire temporaire appelée accumulateur ou registre.

> A cette occasion, on peut préciser que les instructions capables d'être traitées par un microprocesseur comportent toujours deux parties : le code opération qui définit l'opération à effectuer et s'il y a lieu l'adresse du second opérande, le premier étant préalablement chargé dans un registre.
>
> Une addition par exemple se traitera de la manière suivante :
>
> — aller chercher à l'adresse X le premier opérande et le stocker dans le registre A ;
> — additionner le contenu du registre A au contenu de l'adresse Y et placer le résultat dans le registre A ;
> — charger le contenu du registre A dans la case mémoire d'adresse Z.
>
> Une instruction BASIC telle que Z = X + Y sera donc transformée en cette suite d'opérations élémentaires compréhensibles par le microprocesseur.

L'unité de commande

Elle est chargée d'extraire et d'organiser le déroulement des diverses instructions, de les faire exécuter par l'unité de calcul et, de manière générale, de faire appel au moment désiré aux diverses ressources nécessaires à la bonne exécution des programmes.

C'est le véritable chef d'orchestre du microprocesseur. L'unité de commande se charge donc de toutes les tâches de communication au sein du micro-ordinateur. Pour cela, il existe divers chemins d'accès aux informations, les *bus*. Suivant la nature des informations véhiculées, on distinguera le bus de données, le bus d'adresse, le bus d'entrée/sortie.

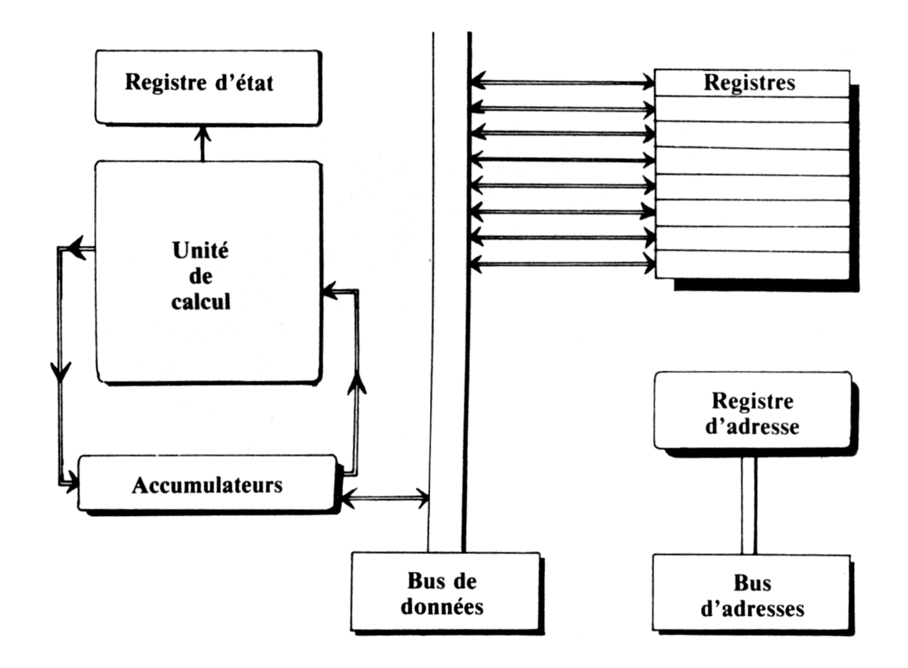
On caractérise également un bus par le nombre de bits qu'il est susceptible de transmettre en même temps, c'est-à-dire par le nombre de fils qu'il comporte. Ces chemins d'accès sont bidirectionnels : c'est l'unité de commande qui, suivant le type d'échange à effectuer (lecture ou écriture), fixera le sens du transfert.

L'unité de commande est reliée à une horloge qui lui délivre des impulsions électriques et qui cadence le fonctionnement du processeur en cycles de longueur fixe. Un cycle correspond à une période d'horloge, c'est-à-dire pour un microprocesseur *à un Mhz*, à 1 microseconde (0,000001 seconde). L'exécution complète d'une instruction comporte toujours un nombre entier de périodes d'horloge, en moyenne deux ou trois.

Les registres

Ce sont de petites mémoires réservées au microprocesseur qui lui permettent de conserver temporairement des informations indispensables à son fonctionnement.

Ils stockent, par exemple, le résultat d'une opération arithmétique, l'adresse de la prochaine instruction à exécuter, l'adresse d'un opérande à *aller chercher* en mémoire...

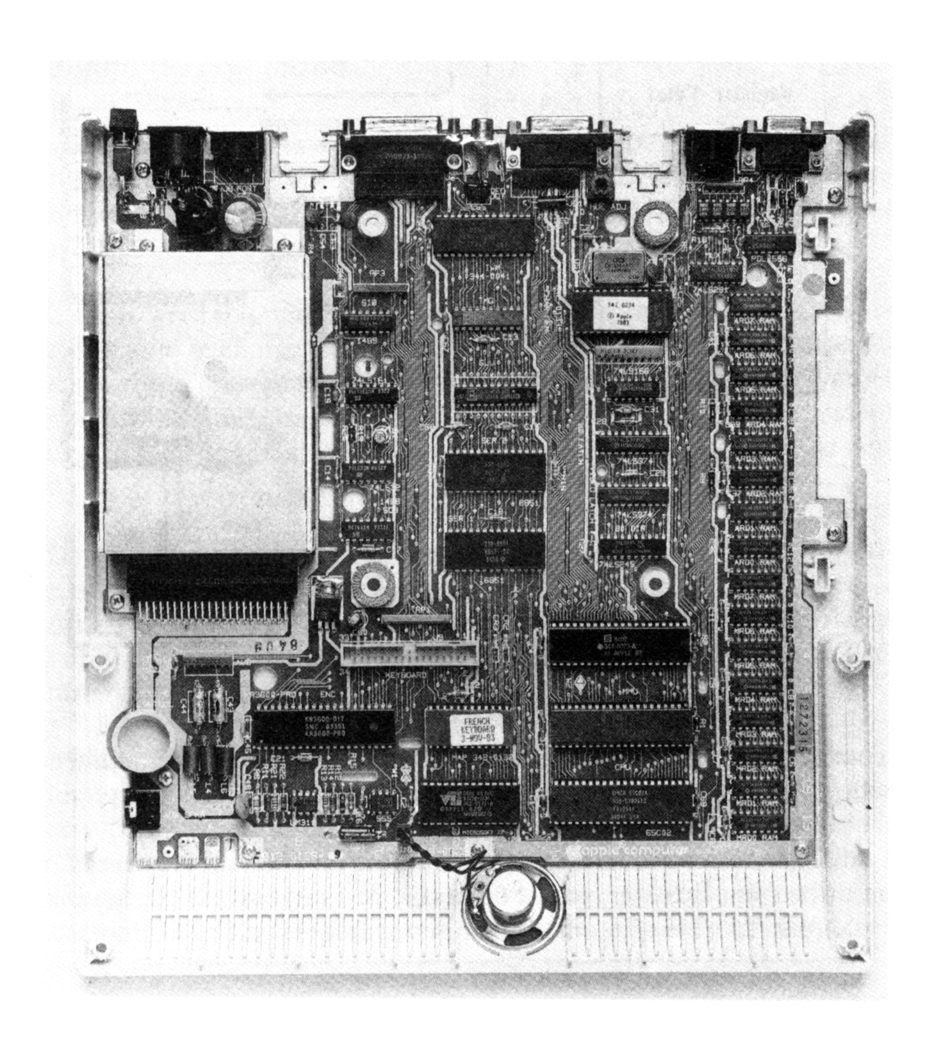


Tous les microprocesseurs sont construits sur cette même architecture.

Ils diffèrent essentiellement par le nombre d'instructions élémentaires qu'ils peuvent exécuter, par la taille des informations traitées en une seule fois, en général 8 ou 16 bits, par la fréquence de l'horloge.

LE MICROPROCESSEUR DE L'APPLE //c : le 65C02

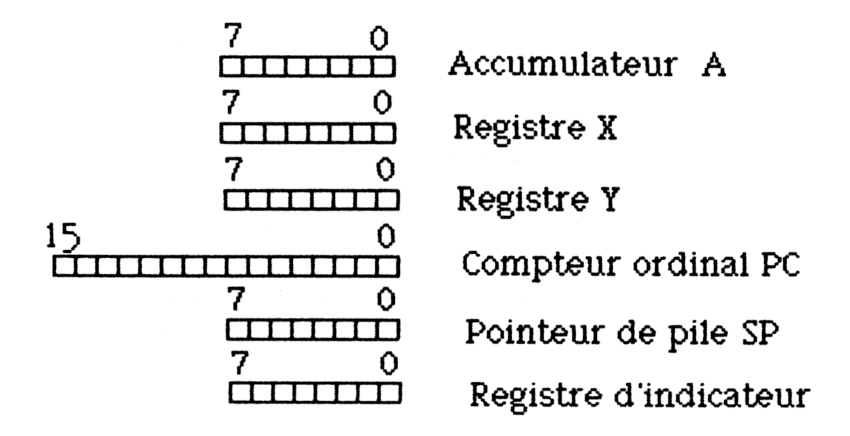
C'est un microprocesseur 8 bits développé par la société Rockwell et appartenant à la famille des "6500", elle-même dérivée des "6800" de la société Motorola. Il est fabriqué en technologie CMOS (Complementary Metal Oxyde Semi-Conductor), ce qui explique la présence de la lettre C dans le libellé. Cette technologie récente permet une grande densité d'intégration et une consommation réduite (ce qui permettrait au 65C02 de fonctionner sur pile).



De plus, les progrès de la technologie CMOS ont permis d'atteindre des vitesses de fonctionnement comparables à celle des technologies réputées plus rapides.

Le 65C02 est une version améliorée d'un microprocesseur de la même firme, le 6502 qui équipait entre autres l'Apple //e. Il reste cependant totalement compatible avec ce dernier, ce qui lui permet de profiter de l'immense bibliothèque de programmes ainsi que de tous les périphériques existants.

Le 65C02 et le 6502 possèdent la même architecture, les mêmes types et nombres de registres :



Le 65C02 possède cependant de nouvelles instructions qui lui procurent des fonctionnalités supplémentaires ; par exemple, effectuer les opérations arithmétiques sur les registres X et Y, et des possibilités d'adressages supplémentaires.

On peut également noter la présence d'instructions de branchements conditionnels que ne possède pas le 6502.

Achevons cette description du microprocesseur de l'Apple //c par une remarque sur la capacité d'adressage du 65C02.

Possédant un bus d'adresse de 16 bits, ce microprocesseur peut théoriquement adresser 2^{16} = 64 Ko. Or l'Apple //c est livré avec une mémoire centrale de 128 Ko.

En effet, par une technique dite de "*bank-switching*" on peut adresser 128 Ko avec un bus de 16 bits.

Il suffit pour cela de faire transférer une partie de l'adresse par le bus de données au moment où il n'y a pas de transfert de données entre le microprocesseur et la mémoire centrale.

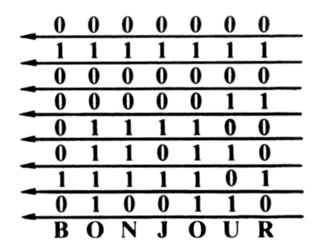
Il faut alors "découper" chaque instruction traitée en plusieurs périodes pendant lesquelles on effectue soit des transferts de données, soit des transferts d'adresses.

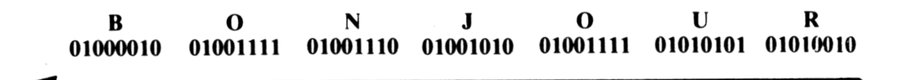
COMMUNICATIONS

Nous avons vu qu'un micro-ordinateur doit pouvoir communiquer des informations à l'extérieur (et en recevoir) par l'intermédiaire de ses périphériques.

Il existe deux types de connexions possibles entre un micro-ordinateur et ses périphériques : les liaisons parallèles et les liaisons séries.

L'unité élémentaire d'information sur un microprocesseur tel que le 65C02 est l'octet. On peut donc transférer cette information soit en un seul paquet de 8 fils (parallèle), soit bit par bit, sur 1 fil (série).

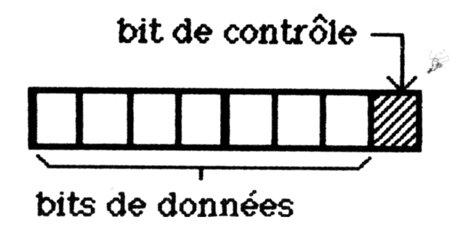




Les microprocesseurs possèdent des ports d'entrée/sortie pour répondre à ces besoins. Ces ports sont de petites mémoires tampons, reliées au bus de données et directement adressables par le microprocesseur.

Les problèmes de fiabilité de la transmission des informations aux périphériques ont rendu nécessaire l'emploi de techniques de contrôle et de correction de ces transferts : lors des opérations d'entrée/sortie, le microprocesseur, suivant un programme spécifique, ajoute à l'information utile, une série d'informations dites de contrôle qui permet, soit de corriger en partie les erreurs, soit d'annoncer à l'utilisateur la présence de ces erreurs.

C'est ainsi que l'on parle de bit de contrôle, de bit de parité, etc.



Lors de la configuration des ports, l'utilisateur peut spécifier le type de contrôle qu'il désire effectuer sur les transmissions.

LA MÉMOIRE

Il existe deux types de mémoire dans l'unité centrale : la mémoire vive, RAM et la mémoire morte, ROM.

Le contenu de la RAM peut être modifié à volonté par l'utilisateur ou par les programmes qu'il exécute, alors que le contenu de la ROM ne peut être modifié.

On comprend alors que les fonctions du système d'exploitation soient stockées en ROM. L'interpréteur BASIC également, ce qui permet de disposer du BASIC dès la mise sous tension de l'ordinateur et de laisser plus de place à l'utilisateur en mémoire vive.

En fait, toute la mémoire vive n'est pas réservée à l'utilisateur : le système y stocke également des informations telles que les adresses utilisées par l'interpréteur BASIC ou des paramètres variables tels que le nombre de caractères affichés par ligne sur l'écran ou le mode d'affichage (normal ou inverse vidéo). Ceci permet à l'utilisateur de modifier ces paramètres en cours de session, ce qu'il ne pourrait faire s'ils étaient stockés en ROM.

L'unité d'information pour l'ordinateur est le bit (*binary digit*). Ce bit est en fait la représentation par 0 et 1 d'un système physique susceptible de prendre deux états stables correspondant à deux états électriques, par exemple 0 et 5 volts.

Les 0 et les 1 forment l'alphabet qui servira à coder toutes les informations nécessaires au fonctionnement de l'ordinateur. C'est sous cette forme que l'information sera codée, mémorisée puis transmise.

Un groupement de huit bits s'appelle un octet. Un octet peut contenir 256 codes binaires différents.

A chaque caractère, numérique ou alphanumérique, que l'on voudra représenter, sera affectée une représentation binaire. Un code de passage entre la représentation usuelle et la représentation binaire est donc nécessaire. L'Apple //c, comme beaucoup de micro-ordinateurs, utilise le code ASCII (American Standard Code for Information Interchange). Avec le code ASCII, chaque caractère alphanumérique peut être représenté sur un octet.

La capacité d'une mémoire se mesure en kilo-octets. Un Ko vaut 1024 octets.

Remarquons qu'en informatique on utilise également le système hexadécimal (base seize) pour représenter les nombres : ce système a été choisi parce qu'il est aisé de convertir en hexadécimal les données codées en binaire et qu'il permet une représentation plus condensée de l'information.

La représentation des nombres est un peu plus compliquée : en effet, pour coder par exemple un entier compris entre −32768 et +32768 il faut 2 octets. On comprend donc que pour stocker des entiers supérieurs ou des nombres réels il faudra disposer de plus de 2 octets (4 en général).

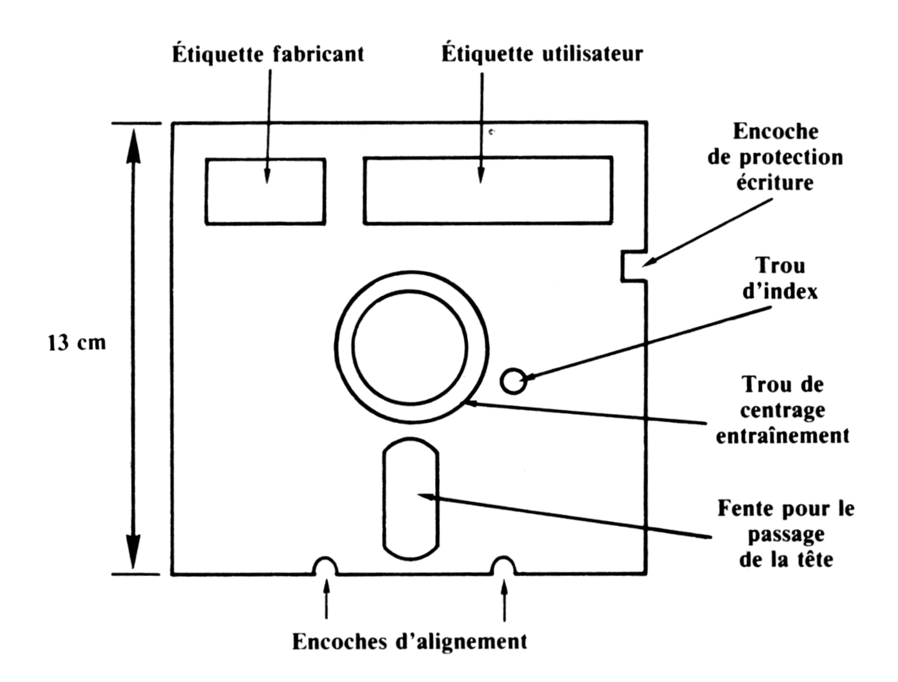
La RAM est divisée en zones mémoire (pages) de taille fixe qui pourront être traitées en bloc.

LES DISQUETTES

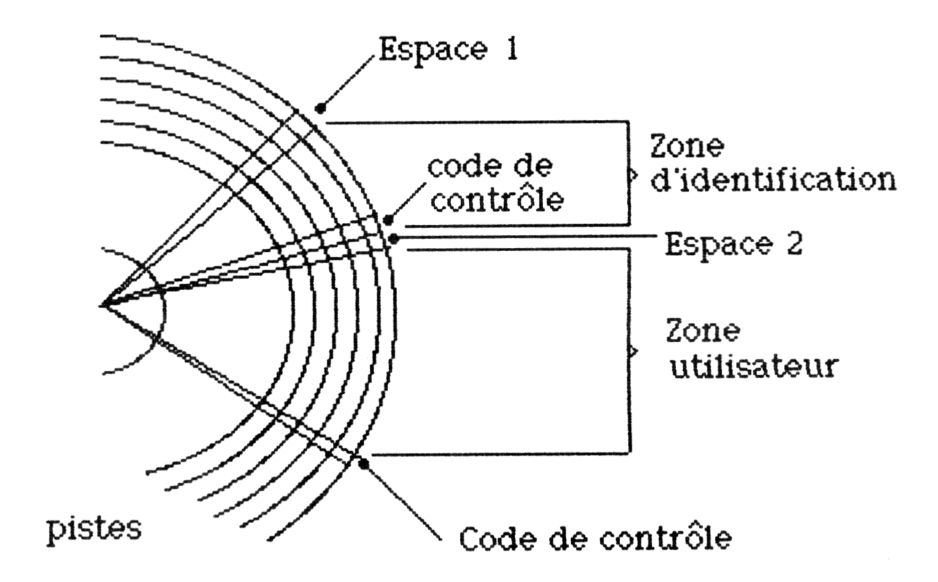
Une disquette, *floppy disk* en anglais pour disque souple, est constituée d'un disque de matière plastique recouvert d'un oxyde magnétique. Il existe des disquettes simple face et des disquettes double face.

La disquette est enfermée dans une pochette de protection laissant apparaître des fenêtres nécessaires à l'entraînement de la disquette et à la lecture des informations.

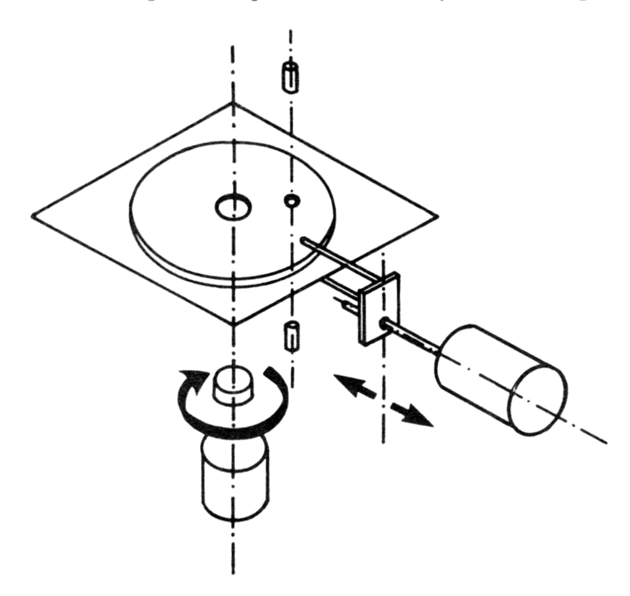
L'Apple //c accepte les disquettes simple ou double face de format 5 pouces 1/4 mais une seule face est accessible à la fois par le lecteur, la face intérieure, opposée à l'étiquette. Le format 5 pouces 1/4 est un standard pour la quasi-totalité des micro-ordinateurs de bureau; exception faite des cousins Macintosh et Lisa qui utilisent des disquettes récentes plus petites (3 pouces 1/2). L'intérieur de la pochette est recouvert d'un matériau limitant les frottements (le Mylar) car la disquette en action tourne à quelque 300 tours/minute.



L'enregistrement s'effectue sur des pistes concentriques à la surface du disque. Le lecteur intégré de l'Apple //c utilise 35 pistes.



Les informations sont codées sous forme magnétique et sont lues et écrites par une tête qui se déplace sur un rayon du disque.



Cette tête peut donc parcourir toutes les pistes de la disquette lorsque celle-ci est en rotation.

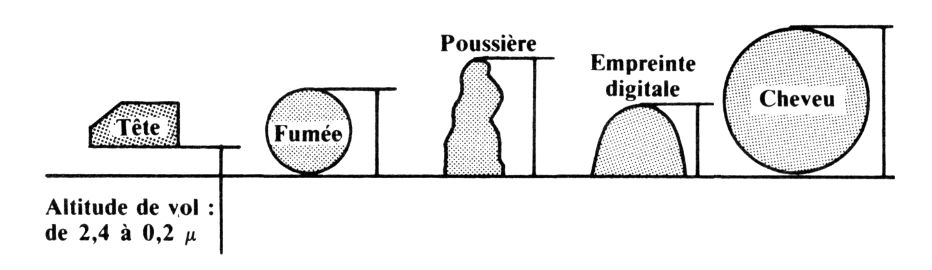
Ce dispositif permet un adressage direct des informations stockées : à la différence d'une bande magnétique, où, pour retrouver une information, il faut parcourir tous les enregistrements précédents, avec un lecteur de disquettes, on peut accéder directement à l'enregistrement recherché. La disquette comporte une piste particulière, reconnue par le système comme étant le catalogue des informations contenues sur les autres pistes.

L'Apple //c est susceptible de stocker 140 Ko (143 360 octets) sur une disquette. Il faut noter que ce n'est pas là une performance exceptionnelle, d'autres systèmes étant capables de stocker plus du double d'informations.

Avant qu'une disquette puisse être utilisée par un micro-ordinateur, elle doit être formatée : un programme spécifique doit dresser une véritable carte de la disquette pour lui permettre d'écrire et de retrouver l'information. Pour cela, il divise chaque piste en 16 secteurs indivisibles ; chaque secteur est repéré par une adresse inscrite sur la disquette.

Il existe deux types de formatage : le formatage physique, qui consiste à faire un trou avant le début de chaque secteur, et le formatage logiciel qui consiste, par programme, à réserver avant chaque secteur une zone où l'ordinateur écrit les informations nécessaires à l'identification du secteur et au contrôle des informations codées.

Sur l'Apple //c, le formatage est logiciel. Il n'existe pas de standard pour l'opération de formatage : une disquette formatée par un type de micro-ordinateur est inutilisable par un autre. Le type de formatage dépend en effet du système d'exploitation implanté sur le micro-ordinateur.



Quelques éléments dangereux pour les disquettes

LE SYSTEME D'EXPLOITATION

QU'EST-CE QU'UN FICHIER ?

L'informatique n'a pas inventé la notion de fichier. Les services administratifs des entreprises le savent bien. Un fichier, de manière générale, est un ensemble d'informations, généralement de même nature et concernant une même application. L'unité élémentaire d'information dans un fichier s'appelle un *article*.

Pour un utilisateur de micro-informatique, quel que soit le type d'application qu'il puisse développer, un fichier est un type de stockage privilégié. Il faut distinguer le fichier et le support sur lequel il est implanté, c'est-à-dire la structure logique, liée au traitement que l'on désire effectuer, et sa structure physique, liée au type de support qui contient le fichier.

Le support est d'ailleurs susceptible d'évoluer dans le temps : un fichier, initialement stocké sur disquettes, peut être transféré en mémoire centrale par l'utilisateur.

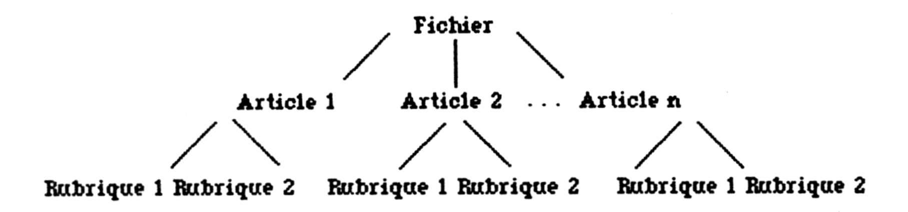
Le support d'un fichier se caractérise par :
— la capacité de stockage ;
— la vitesse de transfert des informations ;
— le temps d'accès.

STRUCTURE LOGIQUE DES FICHIERS

C'est elle qui intéresse l'utilisateur, puisqu'elle conditionne en grande partie leur manipulation.

L'article porte alors le nom d'*enregistrement logique*. Il est décomposé en différentes rubriques pouvant contenir des chaînes de caractères de longueur variable.

Exemple de structure



Exemple d'article

Nom Etudiant	20 caractères
Numéro S.S.	13 caractères
Age	2 caractères
Adresse	30 caractères

STRUCTURE PHYSIQUE DES FICHIERS

C'est elle qui rend compte de l'emplacement sur le support des informations stockées. Elle est imposée à l'utilisateur par le système de gestion de fichiers implanté sur l'ordinateur (système d'exploitation).

Elle est également fonction du type de support utilisé : les données stockées sur disquettes ne le sont généralement pas de la même manière que sur une bande.

L'utilisateur n'a pas à connaître cette structure. Le logiciel de gestion de fichiers convertit lui-même les commandes de l'utilisateur, lecture/écriture d'un article repéré par un numéro, en ordre de lecture/écriture à une adresse calculée à partir de ce numéro. Comme on parle d'*enregistrement logique*, on parle également d'*enregistrement physique*.

En effet, pour diminuer le temps d'accès aux informations le système de gestion de fichiers regroupe dans un même bloc plusieurs enregistrements logiques. Ces blocs d'informations portent le nom d'enregistrements physiques.

LES DIFFÉRENTS TYPES DE FICHIERS

On classe généralement les fichiers par modes d'accès.

Fichier séquentiel

Le stockage des informations s'effectue au fur et à mesure de leur arrivée : une information ne peut être retrouvée qu'après lecture de toutes les précédentes.
Exemple : les fichiers stockés sur bande magnétique sont exclusivement de type séquentiel.

Fichiers à accès direct

A chaque article composant le fichier est attribué un numéro d'ordre par lequel on accède directement à l'article. Ces fichiers imposent un support de type disque ou disquette magnétiques. C'est le cas de l'Apple //c. Le temps d'accès est très court (quelques centièmes de seconde).

Fichiers à accès indexé

La recherche des articles s'effectue à l'aide d'une table de correspondance (index) qui associe à chaque article un numéro d'accès. Cette table de lecture rapide permet une meilleure gestion des modifications (suppressions/insertions) apportées au fichier. Cette gestion est totalement prise en charge par les programmes système.

Index

Rose	1
Tulipe	2
Iris	3

Fichier à accès séquentiel indexé

Ce sont des fichiers de type séquentiel ayant en plus un certain nombre de repères permettant de se positionner directement à des endroits proches de l'information.

QU'EST-CE QU'UN SYSTÈME D'EXPLOITATION ?

Le système d'exploitation peut être considéré comme l'ensemble des programmes nécessaires au "bon fonctionnement" d'un ordinateur, c'est-à-dire à une utilisation simplifiée et agréable pour l'utilisateur. En effet, pour qu'une machine soit utilisable, elle doit prendre en charge un certain nombre de tâches qui constituent le logiciel de base. Nous allons préciser cette notion. On peut distinguer deux types de logiciels :

— le logiciel d'application qui rassemble les programmes que vous écrivez ou que vous achetez tout faits (les progiciels) ;

— le logiciel de base qui est l'outil indispensable au fonctionnement de l'ordinateur. Au sein du logiciel de base il est habituel de distinguer les *programmes utilitaires* et le *système d'exploitation* proprement dit.

LES PRINCIPAUX PROGRAMMES UTILITAIRES

L'éditeur de texte

L'éditeur de texte est un programme qui rend possible l'édition de vos fichiers sur le moniteur. Il permet aussi toutes les modifications (suppression, insertion, correction) que vous voulez apporter à la page que vous tapez, que ce soit un programme, des données numériques ou du texte. Il se comporte donc comme un interface entre l'utilisateur et la mémoire du micro-ordinateur.

L'éditeur est particulièrement important car le confort d'utilisation de votre micro-ordinateur lui est intimement lié : de la qualité de ce programme système dépend la souplesse des manipulations à l'écran.

Remarque : l'éditeur variant avec le système d'exploitation, les conditions d'édition d'un programme BASIC seront différentes de celles d'un programme PASCAL.

Le compilateur

Ce programme traduit les programmes sources écrits par l'utilisateur (en langage évolué) en programmes objets écrits en langage machine (suite de 0 et de 1). Il vérifie la syntaxe de vos programmes, crée les tables nécessaires à la gestion de la mémoire pour l'exécution du programme, génère le "code machine" compréhensible par la machine.

La compilation (traduction du programme source) n'intervient qu'après l'écriture totale du programme de l'utilisateur. Un programme compilé n'est exécutable qu'après sa traduction complète en langage machine.
Sur l'Apple //c, le langage de programmation PASCAL est l'exemple classique du langage compilé.

L'interpréteur

A la différence du compilateur, ce programme envoie immédiatement les lignes d'instructions traduites au microprocesseur ; celui-ci les exécute sans garder trace du code (suite de 0 et de 1). Toute nouvelle exécution demandera une retraduction ; la différence de conception

entre un compilateur et un interpréteur explique les écarts de rapidité d'exécution (un programme compilé s'exécute beaucoup plus rapidement qu'un programme interprété). Notons également que l'interpréteur doit être présent dans la mémoire centrale durant toute la frappe du programme source, ce qui n'est pas le cas du compilateur.

Sur l'Apple //c, l'interpréteur BASIC est résident : il est stocké en ROM et n'a donc pas besoin d'être chargé en mémoire ; il occupe une partie de la mémoire interdite à l'utilisateur.

Si vous voulez programmer en PASCAL, il vous faudra acheter une disquette contenant le compilateur PASCAL puisque celui-ci n'est pas résident.

Divers utilitaires

On regroupe sous ce nom tous les programmes se trouvant sur la disquette "utilitaires-système" !

Ils vous permettent de travailler sur les disquettes (copie, formatage, vérification du bon état de vos disquettes...) et sur les fichiers (destruction, changement de nom, catalogue...).

Ces programmes ne sont pas résidents : sinon ils occuperaient trop de place en mémoire et ne vous permettraient pas d'écrire vos propres programmes ! C'est pourquoi ils sont livrés sur une disquette à charger dès que vous avez besoin d'une de ces options.

Les possibilités offertes par les utilitaires-système sont détaillées dans le manuel d'utilisation de l'Apple //c. Nous en expliciterons le fonctionnement après quelques rappels sur les fonctions d'un système d'exploitation.

LE SYSTÈME D'EXPLOITATION

(On trouve encore souvent l'abréviation O.S. venant de l'anglais "Operating System".)
Le système d'exploitation rassemble toutes les tâches devant assurer la fonction d'interface entre les programmes de l'utilisateur et le matériel que constitue l'ordinateur.

Cet ensemble de petits sous-programmes, appelés parfois fonctions ou primitives, réside en ROM. C'est à partir de ces primitives que travailleront les programmes livrés avec la disquette utilitaires-système.
Exemples de quelques primitives : la sauvegarde et l'effacement d'une zone mémoire, l'écriture à l'écran du contenu d'un fichier.
Du point de vue de l'utilisateur, le système d'exploitation est perçu comme un ensemble de commandes généralement composées au clavier. Ces commandes peuvent également être *intégrées* aux programmes d'applications : par exemple, les ordres de lecture ou d'écriture dans des fichiers. En fait, le système d'exploitation se charge de bien d'autres tâches.

• Tout d'abord, il effectue l'initialisation de l'ordinateur à la mise sous tension et lors des "RESET" effectués par l'utilisateur. Ce travail consiste principalement à configurer la mémoire et l'ensemble des ports d'entrée/sortie afin que la machine soit prête à recevoir les instructions.

• Il doit d'autre part se charger de la gestion des périphériques (affichage sur l'écran, lecture des caractères composés au clavier, pilotage d'une imprimante, d'un lecteur de disquettes...).

• Il doit enfin s'occuper de la gestion de la mémoire (gestion des variables définies par l'utilisateur, réservation de zones, déplacement éventuel de ces zones en cours d'utilisation, transferts entre la mémoire centrale et les disquettes...).

Le véritable cerveau du système d'exploitation s'appelle le moniteur : il s'occupe plus particulièrement de l'enchaînement des tâches. Un programmeur BASIC curieux peut le rencontrer à l'occasion d'une exploration de la mémoire : si, par hasard, il vient à rompre le fil directeur d'un programme en allant écrire dans certaines cases mémoire, le micro-ordinateur s'affolera lors de l'exécution. La procédure prévue alors par les concepteurs de l'Apple //c consiste à annuler les opérations en cours sans redonner la possibilité au programmeur de revenir au BASIC. Le curseur change alors de forme : une étoile apparaît sur l'écran ; elle signifie que le micro-ordinateur ne comprendra plus désormais les instructions formulées en BASIC. Vous êtes en dialogue direct avec le moniteur, mais attention : il parle en hexadécimal. Un "CONTROL-RESET" vous ramènera dans le BASIC connu.

Dans le cas d'un système multi-utilisateurs, le système d'exploitation doit également s'occuper de l'ordonnancement des différentes tâches demandées par les utilisateurs : en effet, il faut pouvoir *partager* correctement l'ordinateur entre les divers programmeurs. Ceci pose de nombreux problèmes tels que la protection des fichiers, l'attribution des ressources (mémoire centrale, périphériques, sous-programmes...) entre les différents utilisateurs. L'Apple //c n'est pas une machine destinée à recevoir plusieurs postes de travail.

PRODOS

Le système d'exploitation PRODOS (Professional Disk Operator System) est le système vendu avec l'Apple //c. Il fait suite au système DOS 3.3 qui équipait les autres modèles de la gamme Apple.

Le DOS 3.3 avait été conçu pour gérer des disques souples de capacité réduite et avait montré ses limites pour la gestion des disques durs et des fichiers de taille importante. C'est pour ces raisons qu'Apple a conçu le système PRODOS qui n'a pas ces inconvénients et qui offre de plus un certain nombre d'améliorations par rapport au système précédent ; nous allons les détailler dans ce chapitre.

Cependant, et ceci est fondamental, le PRODOS offre *une certaine compatibilité* avec le DOS 3.3 : concrètement, vous pouvez profiter de l'immense bibliothèque de progiciels sous DOS 3.3. Nous reviendrons plus en détail sur cette compatibilité PRODOS-DOS 3.3. Aucune compatibilité avec les versions antérieures du DOS n'est annoncée. (Avec le DOS 3.2 en particulier.)

FONCTIONNALITÉS DU PRODOS

Comme le DOS 3.3, le PRODOS est un système capable de gérer deux lecteurs de disquettes en plus des tâches habituelles d'un système d'exploitation.

Il est capable d'adresser 32 mégaoctets de mémoire comme une seule unité logique ; en effet, alors que le DOS 3.3 utilise un adressage physique des disquettes, PRODOS emploie un adressage logique : il

utilise le nom d'accès défini par l'utilisateur (qui correspond au nom de la disquette) et non le numéro du lecteur où se trouve la disquette. Ainsi, si vous possédez deux lecteurs de disquettes, lors d'une recherche de fichiers, vous n'aurez pas à spécifier sur quel lecteur faire la recherche : si les deux disquettes portent le même nom, PRODOS pourra les considérer comme étant une seule unité logique ; il ira rechercher le fichier dans la première disquette et s'il ne le trouve pas, continuera ses recherches dans la deuxième disquette.

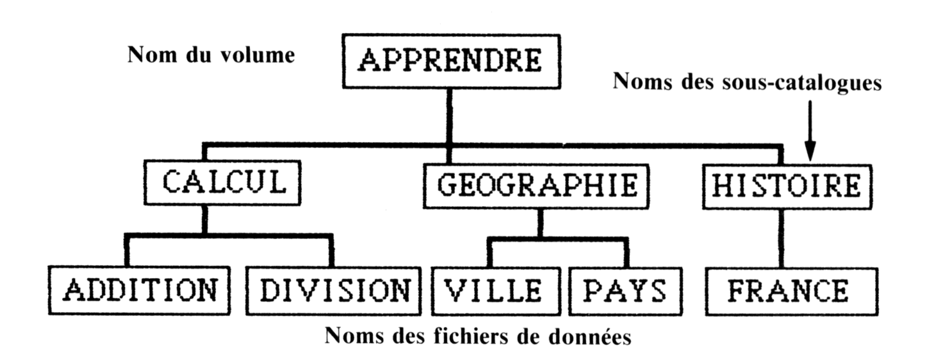
On pourra d'autant plus regretter la faible capacité des disquettes employées (140 Koctets).

Ces performances sont autorisées par le type d'adressage utilisé par le PRODOS : alors que le DOS 3.3 repère les informations stockées sur disquette par leur adresse physique (numéro de secteur), PRODOS stocke les fichiers grâce à des blocs logiques de 512 Koctets.

La représentation logique des informations stockées est donc indépendante de leur implémentation physique sur le disque. Comme nous allons le voir, cette indépendance permet au PRODOS d'être très performant dans la gestion des fichiers.

PRODOS utilise un système déjà éprouvé par UNIX, qui donne à l'utilisateur la possibilité de créer des *sous-catalogues* en plus des catalogues de chaque disquette. Chaque sous-catalogue peut contenir n'importe quel type de fichier, y compris des fichiers sous-catalogue.

Exemple de structure



Parcours possibles : APPRENDRE/CALCUL/ADDITION
APPRENDRE/CALCUL/DIVISION
APPRENDRE/GEOGRAPHIE/VILLE
APPRENDRE/GEOGRAPHIE/PAYS
APPRENDRE/HISTOIRE/FRANCE

A chaque instant, pour spécifier le sous-catalogue accessible, PRODOS utilise des *préfixes*, c'est-à-dire des mots qui lui permettent de rechercher le fichier demandé dans le bon sous-catalogue.

Ainsi, pour exécuter un programme nommé ADDITION rangé dans le sous-catalogue CALCUL qui se trouve lui-même dans une disquette s'appelant APPRENDRE, vous devrez taper :

RUN/APPRENDRE/CALCUL/ADDITION.

Pour alléger cette écriture, il est possible de fixer momentanément ce préfixe en RAM grâce à une option proposée par la disquette utilitaires-système (voir p. 48).

Pour pouvoir stocker des fichiers allant jusqu'à 16 mégaoctets, PRODOS utilise des méthodes bien connues d'adressage indirect par index.

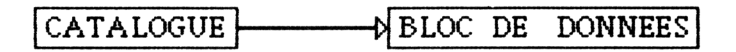
— Pour un fichier de taille inférieure à 512 octets, le système affecte simplement un bloc et un numéro qu'il saura retrouver.

— Pour un fichier de taille comprise entre 513 octets, et 64 Koctets, PRODOS va créer un index (stocké naturellement dans un bloc) qui archivera les 256 numéros possibles des blocs, qui eux contiendront les données.

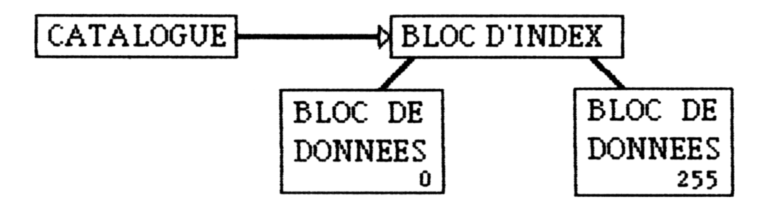
— Pour des fichiers de taille supérieure, PRODOS créera deux index : le premier contiendra les 128 numéros possibles des blocs qui contiendront les index ; ces index adresseront les 256 blocs éventuels qui stockeront les données.

On pourra stocker ainsi des fichiers allant jusqu'à 16 mégaoctets. Il est à noter que ce procédé est totalement transparent à l'utilisateur : lors des mises à jour, le système effectuera automatiquement les opérations nécessaires.

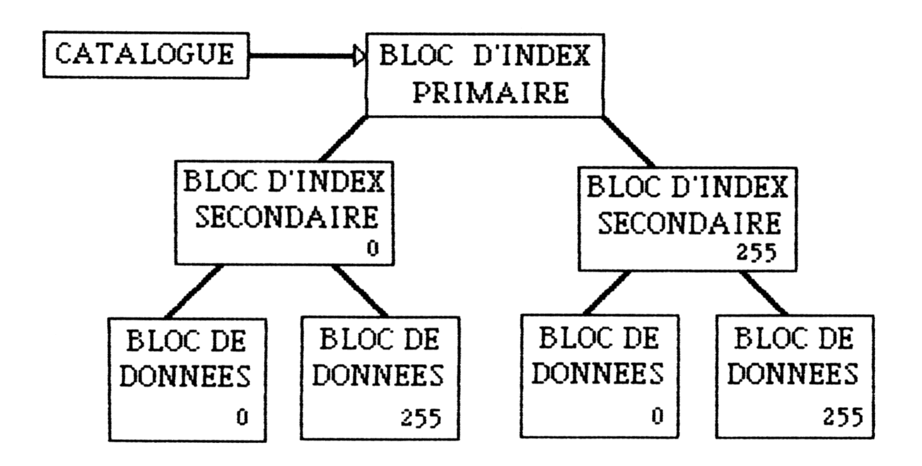
• **de 0 à 512 octets**



• **de 513 octets à 128 Ko**



• **de 129 Ko à 16 mégaoctets**



Méthodes de stockage des fichiers

Pour l'utilisateur *moyen*, PRODOS est en tous points similaire au DOS 3.3. Il permet à l'utilisateur de se décharger des tâches de gestion de fichiers et de périphériques en toute tranquillité.

PRODOS comporte cependant, par rapport au DOS 3.3, un certain nombre de commandes supplémentaires. Par exemple, il est possible de reconfigurer les ports série pour utiliser des périphériques non spécifiques à la marque Apple. Il existe d'autre part une routine système qui vérifie le bon état d'une disquette.

La structure particulière de PRODOS ne lui permet pas d'être totalement compatible avec le DOS 3.3 : ainsi, PRODOS n'est pas capable de lire des fichiers écrits sous DOS 3.3. Cependant, grâce à un programme existant sur la disquette utilitaires-système, il est possible de transférer des fichiers de DOS 3.3 en PRODOS ; il est hélas ! possible qu'il faille effectuer des modifications sur les programmes écrits sous DOS 3.3 pour qu'ils puissent être exécutés sous PRODOS.

Par exemple, les programmes appelant certaines primitives du système, existant sur DOS 3.3 et non sur PRODOS, seront particulièrement candidats pour *planter le système* ! Il est toujours possible de s'en sortir, mais parfois au prix de nombreuses difficultés. Remarquons, tout de même, que la plupart des conversions se passent sans trop de problèmes.

Selon Apple, le DOS 3.3 effectue les transferts de données (écriture ou lecture sur une disquette) à une vitesse de l'ordre de 1 Ko par seconde, tandis que PRODOS effectue la même opération à une vitesse de 8 Ko par seconde, ces chiffres étant donnés pour l'Apple //e et ne tenant pas compte des accès aux *directories* ou d'éventuels échanges de données avec le BASIC.

On peut, en tout cas, estimer que le système PRODOS améliore la vitesse des échanges entre la mémoire centrale et les disquettes dans un rapport au moins égal à 5.

LES UTILITAIRES-SYSTÈME

Après avoir chargé cette disquette, apparaît sur l'écran le menu principal qui permet à l'utilisateur de sélectionner le programme qu'il désire exécuter.

```
Utilitaires Système          Menu Principal
  Copyright Apple Computer, Inc., 1984
_______________________________________________

Travail sur Fichiers

1. ‹ COPIER FICHIERS ›
2.   Effacer Fichiers
3.   Renommer Fichiers
4.   Verrouiller/Déverrouiller Fichiers

Travail sur Volumes

5.   Copier Volume
6.   Formater Volume
7.   Identifier/Lister Volume
8.   Autres Activités

9.   Quitter les Utilitaires Système

Taper un nombre ou presser ↓ ou ↑ pour
sélectionner une option. Presser ↵_
_______________________________________________
Aide : Presser  -? ou  -?
```

L'ensemble des programmes est alors présent en mémoire centrale. On peut retirer la disquette du lecteur.

Cette disquette contient une série de programmes destinés à aider l'utilisateur à gérer efficacement l'ensemble de ses disquettes.

Par exemple, on y trouve les programmes permettant la copie de diquettes et de fichiers, le formatage ainsi que d'autres opérations dont nous verrons le détail au cours de ce chapitre.

Ces programmes utilisent des fonctions stockées en mémoire morte (ROM) dans l'ordinateur. Il faut bien comprendre qu'il existe déjà "dans" l'ordinateur des programmes utilitaires tels que ceux de création ou de sauvegarde de fichiers. Il faut bien qu'un "service minimum" soit toujours assuré dès l'initialisation du micro-ordinateur.

Il est important de noter qu'une bonne connaissance du fonctionnement de tous ces programmes est une condition nécessaire à une bonne utilisation de votre micro-ordinateur. Avant toute chose, il convient de se familiariser avec le maniement de cette disquette. Nous tâcherons donc, sans reprendre le manuel d'utilisation, de vous détailler le rôle et le fonctionnement des programmes proposés par la disquette utilitaires-système.

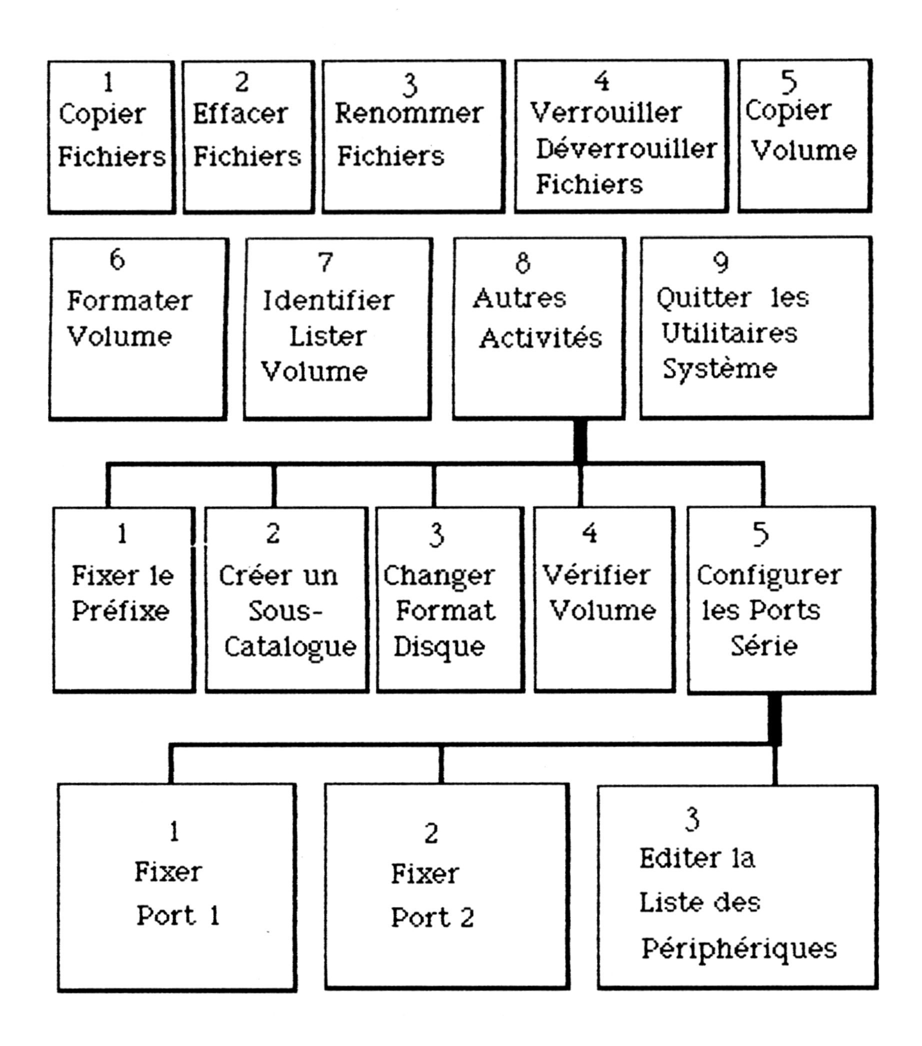
Huit programmes sont proposés à l'utilisateur : quatre concernent des travaux relatifs aux fichiers et quatre autres se rapportent aux disquettes. Une neuvième option permet de sortir des utilitaires-système, mais sans retrouver l'espace de travail existant avant chargement. Par exemple, le chargement de la disquette utilitaires-système effacera de la mémoire les programmes qui s'y trouvent.

Notons tout de suite qu'il est toujours possible et conseillé d'obtenir des renseignements complémentaires sur le déroulement des programmes en utilisant la touche "HELP" ("POMME OUVERTE" et "?" sur l'Apple //c) : ceci provoquera l'affichage sur l'écran d'un petit texte explicatif sur la marche à suivre. Cette assistance est accessible à tout moment.

De plus, si en cours d'utilisation vous désirez abandonner le déroulement du programme en revenant au menu principal, vous pouvez actionner la touche "ESCAPE".

Remarque

Certaines des options proposées font appel à la notion de *nom d'accès PRODOS* ; cette notion étant un peu délicate, nous n'en parlerons qu'en fin de chapitre. Cet outil, bien que fort utile pour le programmeur expérimenté, peut être délaissé dans une première approche par le programmeur débutant.



Menu Principal des utilitaires système
1 Copier Fichiers
2 Effacer Fichiers
3 Renommer Fichiers
4 Verrouiller Déverrouiller Fichiers
5 Copier Volume
6 Formater Volume
7 Identifier Lister Volume
8 Autres Activités
9 Quitter les Utilitaires Système
1 Fixer le Préfixe
2 Créer un Sous-Catalogue
3 Changer Format Disque
4 Vérifier Volume
5 Configurer les Ports Série
1 Fixer Port 1
2 Fixer Port 2
3 Editer la Liste des Périphériques

Formatage d'une disquette

Cette opération prépare une disquette afin de lui permettre de recevoir des informations. Cela consiste à délimiter sur les pistes magnétiques de la disquette des secteurs physiques, qui serviront à stocker et à repérer vos fichiers sur le volume. Le formatage est une opération indispensable avant toute utilisation d'une disquette : une disquette non formatée est inutilisable.

Notons que cette opération est spécifique du système d'exploitation : une disquette formatée par un système est généralement inutilisable par un autre système. Il n'existe pas de standard sur lequel les constructeurs se soient mis d'accord. A l'intérieur d'une même gamme de micro-ordinateurs, on peut trouver quelquefois une certaine compatibilité ascendante : le nouveau système étant capable de lire les disquettes formatées par l'ancien sans autre intervention cabalistique du programmeur.

Le formatage détruit toutes les informations qui existaient précédemment sur le disque : une disquette nouvellement formatée est totalement vierge pour l'utilisateur.

Le programme de formatage est obtenu en sélectionnant l'option 6 du menu principal ; il apparaît alors un menu secondaire associé à cette opération.

Il suffit ensuite de préciser à la machine l'endroit où se situe la disquette à formater (lecteur intégré ou lecteur externe) et à confirmer cette requête.

Le programme vous permet également de donner un nouveau nom à votre disquette. Si vous ne le faites pas, le programme s'en chargera lui-même ! Cependant, pour des opérations un peu complexes que nous verrons par la suite, vous avez tout intérêt à lui donner vous-même un nom ; vous pourrez ainsi vous en souvenir plus aisément.

Identification d'un volume

Ce programme permet d'afficher sur une imprimante ou sur votre moniteur la liste des fichiers existant sur une disquette ainsi que leur type (BASIC, texte, binaire...) et la "place" qu'ils occupent.

On obtient également le nom de la disquette et le nom du système qui l'a formatée (PRODOS, DOS 3.3).

La place occupée par l'ensemble des fichiers est donnée en secteurs ou en blocs suivant le système d'exploitation utilisé. En effet, le PRODOS *divise* la disquette en 274 blocs élémentaires et le DOS 3.3 en 496 secteurs.
Ces blocs ou secteurs sont l'unité élémentaire d'enregistrement sur la disquette. Ils sont le plus petit espace mémoire que l'ordinateur peut traiter sur une disquette (plus petite unité adressable).

Ce programme est sélectionné par l'option 7 du menu principal.

Renommer un fichier

Cette option vous permet de changer le nom d'un fichier que vous avez créé et stocké sur une disquette.

Ce peut être utile pour sauvegarder plusieurs versions d'un programme ou d'un fichier. En effet, si vous apportez des modifications à un fichier et le sauvegardez sur disquette sous le même nom, vous perdrez la version précédente de votre travail : au moment du chargement sur la disquette, la nouvelle version ira *écraser* la précédente. En supposant que vous vouliez conserver plusieurs versions d'un travail, il vous suffit donc de changer le nom de la première version puis de sauver sur disquette la nouvelle en lui affectant l'ancien nom de la première version : ainsi, le dernier travail effectué portera toujours le même nom, et c'est très commode.

La marche à suivre est tout à fait similaire à celle des programmes précédents. Il suffit de vous laisser guider par la machine !
Cette option porte le numéro 3 dans le menu principal.

Verrouiller et déverrouiller un fichier

Il s'agit ici de proposer à l'utilisateur un certain nombre de sécurités pour lui éviter d'effacer accidentellement un fichier. Verrouiller un fichier consiste à *inscrire* sur la disquette où il est stocké, dans une zone réservée à cet effet, une information lue par le système avant toute opération d'écriture ; cette information interdira toute modification du contenu du fichier.

Pour modifier un fichier verrouillé, il faudra alors le déverrouiller, c'est-à-dire supprimer cette information. Il faut notei que la protection n'est pas complète. En effet, lors du formatage d'une disquette, le système ne vérifie pas son contenu et écrase systématiquement tous les fichiers, verrouillés ou pas.

Pour effectuer cette opération il vous suffit de sélectionner l'option 4 du menu principal et le lecteur de disques utilisé. Le programme affiche alors à l'écran le nom de tous les fichiers existant sur la disquette ; il suffit d'indiquer les fichiers à verrouiller parmi la liste proposée.

Effacer un fichier

Ce programme vous permet d'effacer un fichier d'une disquette. Avant d'effectuer cette opération, il est recommandé de lister les noms de tous les fichiers existant sur la disquette à l'aide de l'option *identification d'un volume*. On évite ainsi des erreurs de noms qui peuvent s'avérer catastrophiques ! Ce programme porte le numéro 2 dans le menu principal.

Recopier un fichier

Cette option vous permet de recopier un fichier d'une disquette sur une autre. Elle est donc indispensable pour effectuer des sauvegardes qui vous permettront de conserver avec plus de sécurité vos fichiers importants.
Pour ce programme, il vous faut préciser le lecteur de disquettes où se situera votre disquette source (celle qui contient le fichier à recopier), ainsi que le lecteur qui contient la disquette destination (celle sur laquelle sera effectuée la copie).

Si vous ne possédez que le lecteur de disquettes intégré, il est tout de même possible d'effectuer la copie : il suffit en effet de placer la disquette source dans le lecteur puis de sélectionner le nom du fichier à recopier. Ce fichier est alors chargé en mémoire centrale ; il faut ensuite placer la disquette destination dans le lecteur ; l'ordinateur recopiera le contenu de la mémoire centrale sur cette disquette. Bien entendu, si vous possédez deux lecteurs de disquettes vous n'avez pas à effectuer cette opération : le transfert se fait automatiquement d'une disquette à l'autre.

Remarque : la place disponible en mémoire centrale n'est souvent pas suffisante pour recevoir le fichier à recopier en une seule fois ; il faut alors placer alternativement les disquettes source et destination dans le lecteur afin d'effectuer la recopie *morceau par morceau.*
Le menu vous indique de façon claire les opérations à effectuer. Il faut cependant prendre soin de ne pas intervertir les deux disquettes sous peine de perdre le contenu du fichier à recopier !

Copier un volume

Il s'agit ici de recopier la totalité des fichiers d'une disquette sur une autre. Le principe de fonctionnement est en tous points similaire au précédent.
Il est à noter que ce programme formate automatiquement la disquette destination avant d'effectuer la recopie dans le même format que celui de la disquette source.

La capacité mémoire d'une disquette étant supérieure à celle de la mémoire centrale, il vous faudra procéder alternativement au chargement des deux volumes si vous ne possédez qu'un seul lecteur de disquettes.

Remarque : une première utilisation judicieuse de ce programme serait d'effectuer une sauvegarde de la disquette utilitaires-système.

AUTRES ACTIVITÉS

C'est l'option 8 du menu principal ; elle recouvre différents programmes que nous allons passer en revue.

Créer un sous-catalogue

Le catalogue d'une disquette est l'ensemble des noms de tous les fichiers qui y sont stockés. Le système PRODOS vous donne la possibilité de subdiviser ce catalogue en plusieurs sous-catalogues, ce qui vous permettra de regrouper vos fichiers en vue d'applications particulières. Pour cela, vous devez créer un fichier de type sous-catalogue à l'aide de ce programme et lui donner un nom.

Vous pourrez ainsi accéder sélectivement à vos fichiers : il faudra alors taper à l'écran le nom de la disquette suivi du nom du sous-catalogue et enfin le nom du fichier que vous recherchez sous la forme suivante :

APPLE/DEUX/C

où APPLE est le nom de la disquette, DEUX est le nom du sous-catalogue, et C est le nom du fichier. Nous verrons, par la suite, comment simplifier cette écriture.

En créant plusieurs sous-catalogues, vous pourrez ainsi appeler plusieurs fichiers par le même nom; il vous suffira de les ranger dans des sous-catalogues différents.

Fixer le préfixe

Nous avons vu dans le paragraphe précédent comment créer un sous-catalogue et comment aller y rechercher ou y sauvegarder un fichier : il est nécessaire en plus du nom du fichier de spécifier le sous-catalogue qui le contient. Pour alléger cette écriture, vous pouvez momentanément fixer en RAM le nom du sous-catalogue utilisé. Ainsi vous n'aurez qu'à préciser le nom du fichier auquel vous voulez accéder, la machine allant elle-même rechercher le libellé du sous-catalogue. Naturellement si vous désirez accéder à des fichiers appartenant à un autre sous-catalogue, il vous faudra changer la valeur du préfixe ou taper le nom complet.

Vérifier les volumes

Cette opération permet de vérifier le bon état d'une disquette. Un programme système va contrôler sur la disquette des informations non accessibles à l'utilisateur mais écrites par le système pour lui permettre d'effectuer ce travail.

Changer le format d'une disquette

Nous avons vu précédemment que le type de formatage dépendait du système d'exploitation employé. Ainsi, par exemple, en travaillant sous PRODOS vous ne pourrez pas utiliser directement des program-

mes stockés sur des disquettes formatées sous DOS 3.3. Il est cependant possible de contourner cette difficulté grâce à un programme de la disquette utilitaires-système.

Cette option permet de convertir des disquettes DOS 3.2 en DOS 3.3, DOS 3.3 en PRODOS, et PRODOS en DOS 3.3. Ce programme utilitaire est donc indispensable si vous désirez faire exécuter à votre Apple //c des programmes écrits sur un autre micro-ordinateur de la famille Apple.

Configuration des ports série

Lors du chargement de la disquette utilitaires-système, un programme fixe automatiquement les paramètres (mode impression ou communication, vitesse de transmission, mise en page pour les imprimantes...) nécessaires aux transmissions du micro-ordinateur avec ses périphériques.

Par défaut, ces ports série sont configurés pour communiquer avec une imprimante Imagewriter et avec un modem à 300 bauds.

Si vous désirez effectuer des transmissions avec des périphériques ne possédant pas ces caractéristiques, vous pouvez, à l'aide de ce programme, modifier les paramètres afin de les rendre compatibles avec ceux de vos périphériques.

Remarque : les caractéristiques d'un périphérique sont rassemblées dans un nombre, le N.I.P. (pour numéro d'identification de produit).

LES PROGICIELS

L'APPLE //c ET LES PROGICIELS

La gamme des micro-ordinateurs Apple est réputée pour le nombre, la qualité et la diversité des logiciels qui ont été développés pour elle. Un particulier, comme une petite entreprise, pourra facilement trouver dans l'immense bibliothèque d'Apple le progiciel qui répondra à ses désirs.

Dans une première approche, on peut classer les différents progiciels en quelques grandes familles.

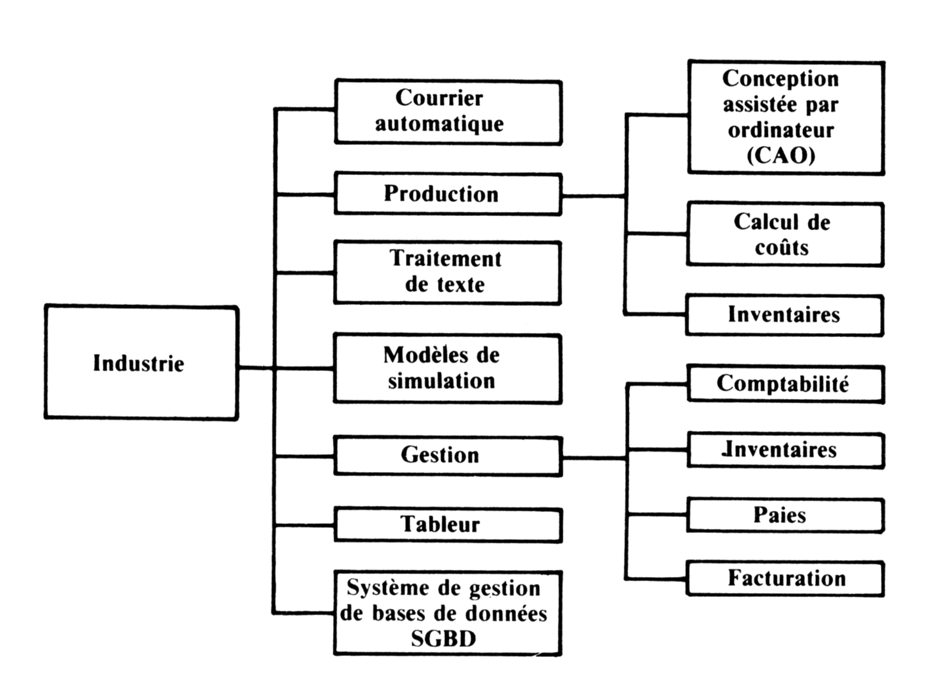
• Les traitements de texte constituent aujourd'hui l'application la plus importante de la micro-informatique dans le travail de bureau.

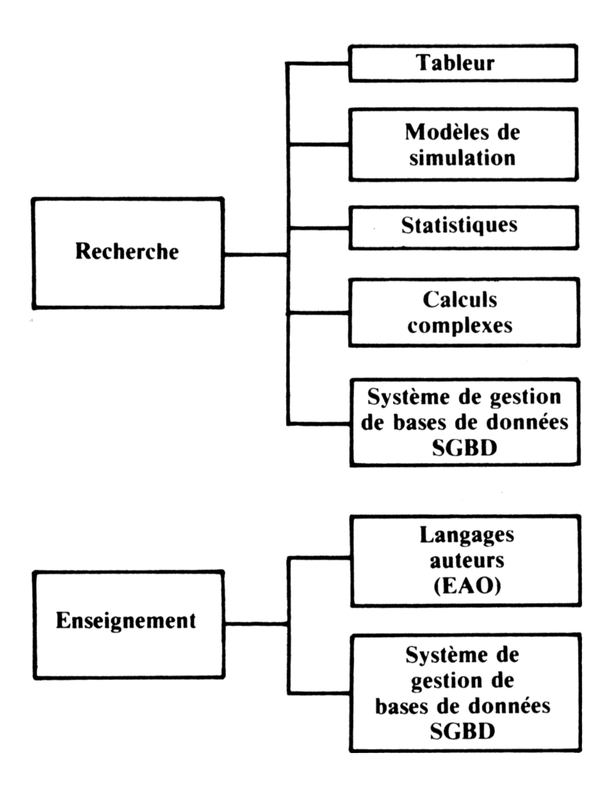
• Les tableurs (appelés aussi "calques") permettent de rassembler des données chiffrées sous forme de tableaux et de les traiter automatiquement à l'aide de formules mathématiques.

• Les logiciels de base de données dont le but est de permettre l'archivage d'un très grand nombre d'informations dans un domaine précis et surtout la récupération rapide et "intelligente" de ces informations. Dans une version moins complète, ces logiciels gèrent des données rangées dans des structures particulières (les fichiers).

• Les "logiciels intégrés", progiciels regroupant sous une même articulation plusieurs logiciels (un traitement de texte, un tableur et un système de gestion de fichiers en général) ; ils sont conçus pour passer très facilement d'une application à une autre sans changer de paramètre de travail.

• Les logiciels de jeux développent les réflexes mais aussi la réflexion (les différentes versions du jeu "Donjons et Dragons" peuvent surprendre... au sens propre comme au sens figuré).





Pour tous les progiciels spécifiques, le choix d'une application va de pair avec le choix du matériel qu'il réclame. En effet, pour être performants, ces logiciels demandent le plus souvent une configuration particulière : extension mémoire, affichage sur quatre-vingts colonnes, manettes de jeux par exemple. L'Apple //c est un système fermé, c'est-à-dire que pour tous les progiciels vendus avec la mention "Apple //c", les extensions mémoire et autres "cartes" nécessaires à l'Apple //e pour bien fonctionner sont ici intégrées dans la machine.

Il reste néanmoins un problème important quand il s'agit d'acheter des périphériques pour compléter votre configuration. C'est le fameux problème de l'interface, un des inconvénients les plus évidents de la micro-informatique. Pour un utilisateur, la conséquence immédiate est qu'il doit s'assurer très précisément auprès de son revendeur que le périphérique qu'il achète est bien compatible avec sa machine. (A ce sujet, voir le chapitre « Périphériques ».)

Une expression classique de la micro-informatique est de dire que tel progiciel fonctionne ‘‘sous’’ tel système d’exploitation. On peut donc dire que le système d’exploitation fait partie de la configuration requise pour l’utilisation d’un progiciel. En conséquence, les fabricants de progiciels actualisent leurs produits au fur et à mesure que les fabricants de machines les rendent plus performantes en améliorant leur système d’exploitation.

Ainsi, on trouve des mentions spéciales sur des progiciels indiquant pour quelles versions ils sont développés ; la ‘‘compatibilité ascendante’’ des systèmes d’exploitation écrits pour la gamme Apple // (DOS 3.3 pour le //e, PRODOS pour le //c) explique que les logiciels vendus pour l’Apple //e puissent fonctionner sur l’Apple //c et non l’inverse. A ce sujet, notons que la mention ‘‘pour Apple //e uniquement’’ figurant sur certains progiciels est mise pour exclure les versions antérieures au //e (comme // + par exemple qui ne dispose pas des touches ‘‘POMME OUVERTE’’ et ‘‘POMME FERMEE’’) mais certainement par pour exclure l’Apple //c.

Remarque : on a déjà observé que le système Apple //c était un système fermé ; il est donc impossible de lui adjoindre la ‘‘carte CP/M’’. En conséquence, les progiciels demandant le système d’exploitation CP/M pour fonctionner ne sont pas utilisables sur l’Apple //c. Ceci concerne surtout les progiciels destinés à la gamme de l’ordinateur personnel d’IBM et ses concurrents compatibles. Un exemple est fourni par le logiciel de traitement de texte ‘‘Wordstar’’. Mais un même progiciel peut fort bien être utilisé sous plusieurs systèmes d’exploitation ; par exemple le tableur ‘‘Multiplan’’ accepte aussi bien CP/M que DOS 3.3.

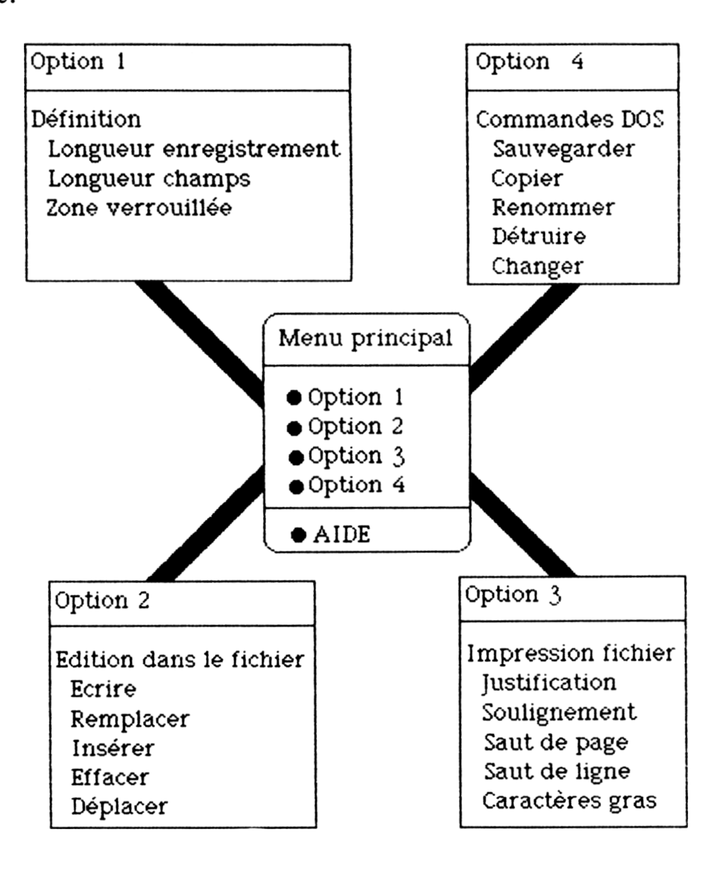
Présentation générale d’un progiciel

Un progiciel se présente à l’utilisateur sous forme d’une ou plusieurs disquettes qui, en général, ne peuvent être recopiées : les auteurs de ces programmes protègent leurs droits. Néanmoins, pour certains produits particuliers, il est prévu de pouvoir copier des parties qui servent souvent et donc s’usent plus vite, les constructeurs ne protégeant que les programmes utiles au moment de la mise en route (ces programmes portent le nom anglais de ‘‘bootstrap’’).

Une fois chargé dans la mémoire centrale du micro-ordinateur, le progiciel fait apparaître un menu, c'est-à-dire un écran sur lequel sont proposées plusieurs options ou ''activités'', et l'utilisateur est amené à en sélectionner une en frappant une touche précise, en déplaçant le curseur avec les touches de direction ou grâce à la souris (voir périphériques).

On retrouve souvent les mêmes types d'options sur tous les progiciels :

• l'édition : écriture, correction, recherche, mise à jour...
• le travail sur les fichiers : définition, suppression, affichage, réorganisation...
• une partie du système d'exploitation : sauvegarde, copies, fonctions spéciales...
• l'impression : définition des paramètres à transmettre à l'imprimante.



LES TRAITEMENTS DE TEXTE

On connaît bien maintenant la réputation des traitements de texte, une des premières applications rendant populaire la micro-informatique.

Le premier intérêt, le plus immédiat, concerne les possibilités très simples et très complètes de corrections. Avec un traitement de texte, on peut considérer que l'on travaille toujours sur la version définitive du texte. Les corrections sont immédiatement prises en compte, les ratures n'existent pas, l'inversion de paragraphes s'effectue à l'aide de quelques touches. Le travail définitif prend forme peu à peu et émerge doucement d'un brouillon qui reste toujours propre.

L'Apple //c est un micro-ordinateur parfaitement adapté au traitement de texte français puisque son clavier offre tous les caractères de la langue française, y compris les minuscules accentuées, le ç et le tréma (c'est ce que l'on appelle un clavier AZERTY accentué complet).

Cette remarque n'est pas superflue car il faut se souvenir que cette machine et les progiciels développés par elle sont conçus pour le public anglo-saxon, au moins dans un premier temps. Il est donc important de s'assurer que les options "françaises" sont accessibles facilement ; à l'usage, il est souvent désagréable d'être obligé de taper les accents circonflexes à côté des lettres sur lesquelles ils doivent se placer (ce point est régulièrement mis en avant par les critiques et est maintenant résolu par certains fabricants).

Les fonctions d'insertion ou de suppression portent généralement sur un caractère, un mot ou un paragraphe, plus rarement sur une phrase. Le programme de traitement de texte se charge à tout instant de décaler le texte déjà écrit pour permettre vos insertions et de combler les espaces laissés vides par des suppressions. Au moment de la frappe, l'utilisateur n'est donc jamais obligé de se préoccuper de la mise en page puisqu'il peut toujours revenir facilement sur son texte, le modifier et en corriger la présentation.

En pratique, ces possibilités de correction ne représentent qu'une commodité et le principal attrait d'un traitement de texte réside dans les fonctions d'impression du programme. Sous cet aspect, ce progiciel est un véritable outil professionnel qui trouve sa pleine mesure dans le domaine de la bureautique pour la duplication de documents ou le courrier personnalisé.

La forme définitive du texte, c'est-à-dire l'aspect qu'il prendra sur le papier, peut être différente de la visualisation qui apparaît à l'écran, car le nombre de colonnes, les décalages, les soulignements et caractères spéciaux n'ont pas les mêmes significations pour le moniteur d'une part et l'imprimante d'autre part.

C'est pourquoi on introduit dans le texte des symboles que le micro-ordinateur n'interprétera pas comme des caractères du texte mais comme des indications à envoyer à l'imprimante en même temps que le texte à lister. Un programme spécial contrôle l'imprimante pour qu'elle effectue les frappes particulières et exécute les ordres de redécoupage du texte (par exemple, lorsque l'on demande un alignement du texte à droite, avec ou sans césure des mots...).

La mise en forme définitive d'un texte s'appelle la justification du texte ou aussi le formatage du texte (à ne pas confondre avec la même expression concernant les disquettes).

La troisième fonction très performante des progiciels de traitement de texte concerne la possibilité d'écrire des programmes traitant des fichiers contenant des textes de la même façon que l'on écrit un petit programme BASIC. L'édition automatique de courrier personnalisé (le terme anglais "mailing" est encore souvent employé) en est l'exemple le plus connu. Schématiquement, il s'agit de travailler simultanément sur plusieurs fichiers contenant chacun des informations devant être utilisées à un moment précis :

— un premier fichier contient des renseignements sur des destinataires (nom, adresse, fonction...) ;

— un second fichier contient des formules appropriées à chaque caractéristique (monsieur, madame, mademoiselle, veuillez agréer...) ;

— un troisième fichier contient le texte original du courrier dans lequel des symboles spéciaux sont frappés en lieu et place des formules particulières et indications personnelles correspondant aux autres fichiers.

Le programme de publipostage (autre nom du courrier personnalisé) doit analyser les symboles spéciaux rencontrés et en conséquence aller explorer les contenus des autres fichiers puis remplacer à l'endroit désigné tous les sigles par les renseignements extraits.

Pour l'utilisateur, un programme de traitement de texte se présente sous la forme d'une ou plusieurs disquettes. Il doit donc charger le contenu de la disquette dans la mémoire centrale du micro-ordinateur. A ce moment, la disquette contenant le progiciel (appelée disquette-maîtresse ou master par opposition aux disquettes sur lesquelles on sauvera le texte tapé) ne sert plus à rien. L'Apple //c a une mémoire suffisamment importante pour qu'un programme de traitement de texte réside en permanence dans sa mémoire centrale.

C'est une des conditions indispensables pour que le progiciel soit facilement utilisable : il ne peut être question de charger régulièrement des "morceaux de programme" au fur et à mesure des besoins.

Après le chargement, la RAM est divisée en plusieurs parties :

• une zone de travail dans laquelle l'utilisateur tape son texte et qui correspond à une zone de fichiers en mémoire pour le micro-ordinateur ;

• une zone réservée au programme de traitement de texte ;

• une zone réservée aux différentes options du programme, constituée de petites mémoires temporaires (appelées tampons ou *buffer*) qui recevront des informations en provenance du programme et les restitueront suivant les instructions de ce même programme (par exemple lors de suppression de caractères ou de déplacement de paragraphe).

Un progiciel de traitement de texte comporte toujours la partie du système d'exploitation nécessaire à l'échange des informations du fichier stocké en mémoire centrale (donc fragile et très sensible aux fausses manœuvres) vers une disquette (lieu plus sûr). C'est ainsi que

l'on trouve toujours les deux opérations qui consistent à sauver et à charger des fichiers ainsi que les manipulations importantes pour le formatage, la vérification des disquettes et les protections (verrouillage et déverrouillage) des fichiers.

DEUX EXEMPLES DE TRAITEMENT DE TEXTE

Le premier, classique et bien connu, conçu pour la gamme des micro-ordinateurs Apple et ayant déjà tourné sur les précédentes versions Apple //+ et Apple //e, s'appelle APPLE WRITER. Le second, plus récent, est un traitement de texte français, EPISTOLE, qui existe sous deux versions dont une particulièrement intéressante pour l'Apple //c puisqu'elle utilise pleinement la souris.

APPLE WRITER

```
Apple Writer //

Copyright 1981-2, Paul Lutus

Copyright 1981-2, Apple Computer Inc

(Pour assistance à l'édition,

pressez Pomme-Ouverte et "?")

Tapez RETURN :
```

```
>Z  Mem:46382 Lng:  501 Pos:     0 Tab:     0 Fich: circulaire2

(Address)

Cher (Title) (Lname):

Félicitations pour votre choix d'un ordinateur Apple. Vous et la famille
(Lname) allez passer des moments agréables et éducatifs avec votre nouvel
ordinateur individuel. Dans un monde moderne en évolution rapide, cher
(Fname), vous ne pouvez vous permettre de vous en passer. Et soyez assuré
qu'avec un Apple, vous avez ce qui se fait de mieux.
Avec nos compliments

L'équipe Apple Computer
```

Ce progiciel a été créé en 1979 et existe actuellement sous plusieurs versions dont la dernière, celle développée pour l'Apple //e, est parfaitement compatible avec le modèle Apple //c. L'affichage apparaîtra sur 80 colonnes puisque la "carte 80 colonnes" est montée en série sur l'Apple //c, et ceci quelle que soit la position du sélecteur 80/40 colonnes. En conséquence, un moniteur est indispensable à l'utilisation de ce logiciel.

Au moment du chargement, APPLE WRITER est contenu sur une seule disquette. Son transfert en mémoire centrale s'effectue en deux temps. En effet, toutes les fonctions d'aide (c'est-à-dire : tableau récapitulatif des commandes, menus, écrans d'explications) ne résident pas constamment dans la mémoire du micro-ordinateur : l'utilisateur les appelle selon ses besoins grâce aux touches spéciales (touche fonction "POMME OUVERTE" et touche "?"). Mais pour ceci, la disquette APPLE WRITER doit se trouver dans le lecteur de disquettes. Ce procédé est courant et permet de ne pas encombrer la mémoire centrale avec des sous-programmes non régulièrement utilisés.

C'est pourquoi, une fois chargée la première partie, on vous demande une première orientation : soit aller à l'"édition" (pour taper du texte), soit consulter le tableau d'aide.

Remarquons que si vous êtes en train de taper un texte et que vous vouliez consulter le menu d'aide, vous devez placer APPLE WRITER dans le lecteur puis charger uniquement les programmes d'aide ; votre texte est conservé durant toutes ces manipulations et vous sera rendu intact.

Plusieurs "écrans" sont possibles avec APPLE WRITER, qui ont chacun une fonction bien déterminée :

• un écran servant à l'édition dans lequel "on tapera tout", c'est-à-dire le texte mais aussi les instructions de mise en page, soulignement... Cet écran est surmonté d'une "ligne de status" donnant à chaque instant des renseignements sur les fonctions utilisées et l'état général du fichier de l'utilisateur (place occupée par exemple) ;

• un écran de visualisation qui présente le texte tel qu'il sera imprimé, donc sans les caractères de contrôle de l'imprimante ;

• un écran temporaire qui agit comme une "fenêtre" donnant la possibilité d'avoir simultanément en mémoire centrale deux fichiers ou deux parties d'un même fichier, et de les visualiser.

L'écran d'édition est fourni par défaut, c'est-à-dire que le programme redonnera toujours la page d'édition après chaque option traitée, alors que les deux autres écrans sont appelés par des commandes.

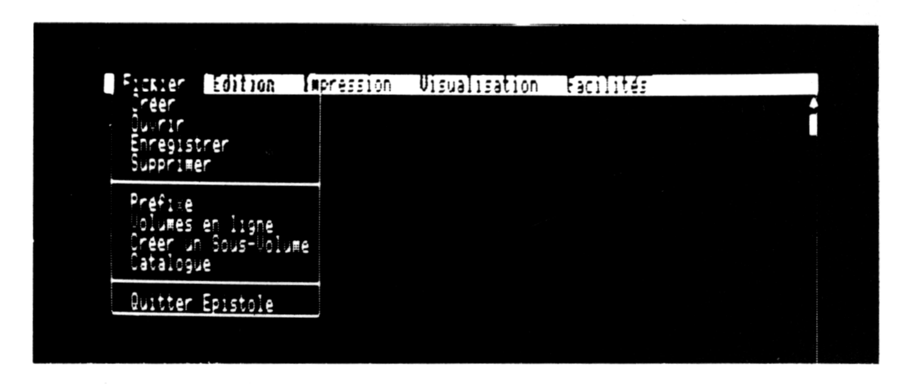
Sous la page d'édition, l'utilisateur dispose d'un certain nombre de commandes permettant d'effacer un caractère, un mot, un paragraphe et éventuellement de les restituer ailleurs; l'insertion, elle, est automatique. Ces commandes sont matérialisées par des lettres mais pour ne pas les confondre avec des caractères du texte tapé, on a recours à un procédé classique en micro-informatique qui consiste à appuyer sur plusieurs touches en même temps : une touche "fonction" et la touche d'un caractère.

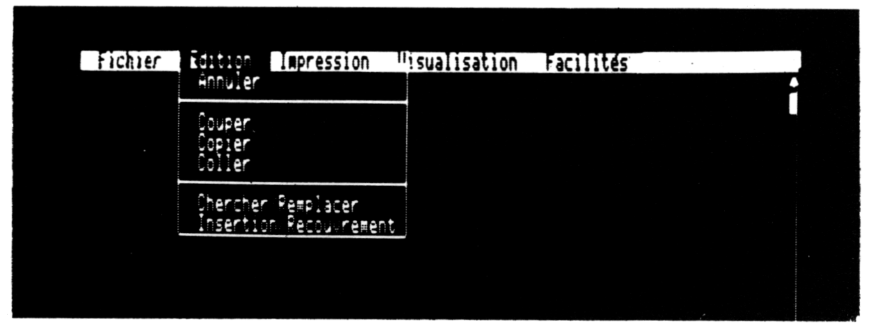
Exemple pour APPLE WRITER :

"CONTROL" et "W" pour agir sur les mots,
"CONTROL" et "X" pour agir sur les paragraphes,
"CONTROL" et "F" pour trouver un mot du texte,
"POMME OUVERTE" combinée aux touches de direction droite ou gauche pour avaler ou restituer des caractères,
"POMME OUVERTE" et "?" pour appeler le menu d'aide (qui contient l'ensemble de ces commandes).

De même, on peut appeler le menu des fonctions d'impression par "CONTROL" et "P" et l'ensemble des commandes du DOS (système d'exploitation de disquettes) par "CONTROL" et "O".

La richesse du progiciel APPLE WRITER oblige l'utilisateur à apprendre un assez grand nombre de codes ou à faire appel régulièrement à la disquette d'aide. On remarquera aussi que les concepteurs de ce logiciel ont attribué des initiales parlantes à ces commandes pour soulager la mémoire de leurs clients (mais attention, c'est en anglais) : F pour *Find*, P pour *Print*, B pour *Begin*, etc.





L'association Apple //c, EPISTOLE, souris permet d'éviter le recours systématique aux touches fonction pour obtenir les commandes du progiciel. L'inconvénient cité plus haut et concernant APPLE WRITER a été levé ici en fournissant à l'utilisateur une page d'édition sur laquelle apparaissent des icônes symbolisant les options disponibles. On accède à ces options en plaçant le curseur sur l'icône correspondante et en validant ; tout ceci est rendu très aisé car c'est la souris qui commande les mouvements du curseur.

Le logiciel EPISTOLE se présente sous forme de trois disquettes :

- une disquette contenant le programme de traitement de texte ;

- une disquette contenant les utilitaires (copie de fichiers, formatage de disquettes...) ;

- une disquette contenant le programme spécifique pour l'utilisation de la souris (version 408).

Remarque : une autre version existe qui n'utilise pas la souris ; on retrouve alors une utilisation en tous points semblable à celle d'APPLE WRITER.

Après chargement, le programme réside en mémoire centrale ; l'utilisateur peut donc enlever la disquette de traitement de texte. Sur l'écran, on trouve le menu principal qui propose quatre options (celles-ci recouvrent le schéma général d'APPLE WRITER) :

• le travail sur les fichiers (création, ouverture) ;

• le travail à l'édition (frappe du texte) ;

• la visualisation (pour voir le document tel qu'il sera imprimé) ;

• l'option des "facilités" qui ouvre sur d'autres menus telles les commandes DOS ou celles d'impression.

La sélection dans les menus, s'opère à l'aide de la souris et donc sans frappe sur le clavier, mais on apprécie pleinement les avantages de la souris sous la page d'édition : en plus des renseignements habituels sur la place mémoire disponible, une série d'icônes (petit dessins) apparaissent en bordure de l'écran. Ces dessins symbolisent les fonctions de suppression, d'insertion, de déplacement, etc. Pour les obtenir, on place le curseur en surimpression et on valide le tout grâce à la souris. De même, le déplacement rapide dans le texte se fait avec la souris placée à un endroit particulier symbolisé par un ascenseur qui "monte" et "descend" au bord du texte.

Cette présentation des fonctions a évidemment l'avantage de dispenser l'utilisateur de tout effort de mémorisation car les icônes sont suffisamment parlantes pour lever toute ambiguïté concernant leur sens.

Pour passer à l'impression, il faut définir la mise en page : justification du texte, type de caractère des titres... Sur ce point, EPISTOLE se présente de façon identique à APPLE WRITER, c'est-à-dire que l'utilisateur frappe dans le texte même les caractères de contrôle pour définir les marges, les paragraphes, le soulignement et l'épaisseur des titres par exemple. La page d'édition présente donc un certain nombre de symboles "indésirables" mais nécessaires à l'imprimante.

Pour palier ce défaut l'option de "visualisation" restitue sur l'écran le document avec les caractéristiques qu'il montrerait s'il était imprimé. Cette possibilité permet de s'assurer de ses ordres de mise en page avant d'imprimer.

Remarque : EPISTOLE ne permet pas d'apporter de modification au texte sous l'écran de visualisation ; suivant l'utilisateur, c'est une sécurité ou un défaut.

EPISTOLE est un progiciel travaillant sous PRODOS alors que APPLE WRITER demande le DOS 3.3. Cette différence est "transparente" à l'usager de l'Apple //c puisque ces deux systèmes d'exploitation ont (également) été écrits dans ce but. Un détail cependant, sans conséquence, attirera l'œil des utilisateurs d'APPLE WRITER : la ligne de status offre parfois des symboles illisibles ; ils correspondent à des emplacements de la mémoire de l'Apple //c prévus pour la souris !

En ce qui concerne les autres possibilités des traitements de texte, comme par exemple le publipostage, les deux progiciels que nous venons de décrire fonctionnent suivant le même principe : à la rencontre de variables (définies précédemment comme telles) dans la page d'édition, le programme va chercher en séquence ce qu'il doit écrire à une place déterminée à l'avance dans un fichier.

Remarquons quand même que le logiciel APPLE WRITER dispose d'un langage de programmation propre appelé WPL avec lequel on peut réaliser des micro-commandes :
— reconnaître les variables d'appel (leur syntaxe) ;
— aller chercher les informations correspondantes dans les fichiers adéquats ;
— les écrire avec le format correct ;
— parcourir une boucle dont l'indice fait défiler en séquence les enregistrements des fichiers concernés.

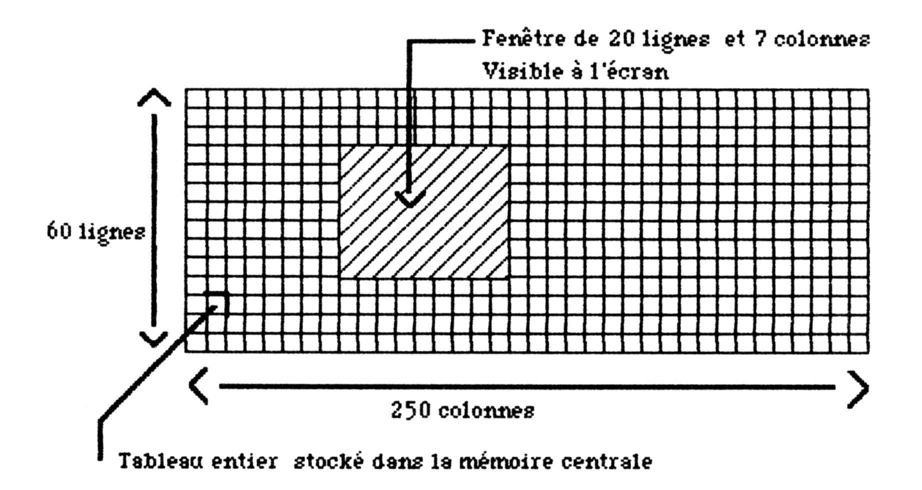
Ce langage de programmation permet d'augmenter les possibilités d'APPLE WRITER, mais évidemment, il demande un petit peu de formation car il s'agit d'écrire des programmes et non plus de se servir d'un progiciel par l'intermédiaire de menus ; de toute façon, les programmes les plus courants (le publipostage par exemple) sont déjà enregistrés et disponibles par simple appel.

Un programme de traitement de texte peut aussi offrir la possibilité de lire dans des fichiers gérés par d'autres progiciels (par exemple aller chercher une partie d'un tableau créé par un tableur) et d'incorporer au texte à éditer les informations ainsi récupérées. EPISTOLE peut le faire avec des progiciels comme MULTIPLAN, VISICALC ou VISIFILE.

La demande de plus en plus grande des utilisateurs vers ce genre de services pousse actuellement les constructeurs à développer les progiciels intégrés (voir p. 77). Notons qu'il existe d'autres progiciels de traitement de texte utilisables sur Apple //c, tel que LE RÉDACTEUR (Sogiciel).

LES TABLEURS*

Le principe des tableurs est particulièrement simple : on imagine un grand tableau de quelques milliers de cases repérées par leur ligne et leur colonne (comme dans le jeu de bataille navale). On ne visualise à l'écran qu'une partie de ce tableau (qui s'appelle une fenêtre) composée de quelques lignes et colonnes contiguës.



*On trouve également l'expression "calc" et sa version francisée "calque" mise pour faire référence à l'option calcul de ces progiciels et aussi parce que le premier tableur s'appelait VISICALC.

Chaque case est susceptible de recevoir des informations :

• soit un texte libellé (un titre, une explication) ;
• soit un chiffre (cadré à volonté) ;
• soit une expression mathématique (en général économique) dont les variables figurent dans d'autres cases.

L'intérêt de la feuille de calcul électronique réside dans l'exploitation systématique de ces formules mathématiques. Le programme se charge de les calculer effectivement et de faire apparaître le résultat de l'opération dans la case correspondante.

Exemple simple : sur un certain nombre de lignes, vous disposez des chiffres correspondant à des recettes ; vous demandez la somme de ces recettes dans la dernière ligne par une formule que vous introduisez à l'emplacement désiré ("somme = recette 1 + recette 2 + recette 3 + ...") et le total apparaît immédiatement.

Recette 1	R1
Recette 2	R2
Recette 3	R3
Total recette	S = R1 + R2 + R3

↑ Colonne des titres (alphanumériques)

↑ Colonne des chiffres (numériques)

Quand R1, R2, R3 sont remplacés par leurs valeurs, S prend la valeur de la somme R1 + R2 + R3.

Pour l'instant, la différence avec un papier comptable et une calculatrice n'apparaît pas clairement. Elle devient évidente si l'on sait que le programme se charge de réagir automatiquement à toute modification dans une donnée et actualise ainsi votre tableau en "temps réel".

Par exemple, le changement d'un chiffre d'une recette entraînera l'ajustement automatique du total ; de même, la modification d'un taux de TVA (une formule parmi d'autres) va se traduire par l'ajustement de toutes les cases où ce taux intervient.

La rapidité de réaction du programme fait son intérêt, la seule peine de l'utilisateur consistant à entrer convenablement les données dans les différentes cases (car la présentation compte beaucoup pour s'y retrouver facilement) et à appliquer les formules d'économie et/ou de mathématique correctes. On peut ainsi calculer des taux de rendement ou d'amortissement, faire des prévisions de remboursement, gérer des portefeuilles d'actions, etc.

Le programme ne vous donne pas ces formules mais une documentation complète existe par ailleurs.

Les principales options d'un tableur se répartissent en groupe de commandes :

• commandes de dialogue entre la feuille de calcul et l'utilisateur ;
— se déplacer d'une ligne, d'une colonne, trouver une case non verrouillée ou non vide...
— effacer des caractères déjà tapés, aller au menu principal...

• commandes du programme lui-même :
— entrée différenciée du texte, des chiffres et des formules ;
— tri de valeurs ;
— opération sur les cases, les lignes et les colonnes (recopie, déplacement, protection) et création de fenêtre;

• fonctions mathématiques (moyenne, écart type, maximum, sinus, racine...) et fonctions particulières (recherche de cases, extraction du contenu d'une case, nombre de caractères d'un libellé...).

De toutes ces applications, l'une des plus importantes est la possibilité de créer plusieurs fenêtres. En effet, on ne peut visualiser à l'écran qu'une partie du tableau et il est parfois très utile d'en juxtaposer différentes parties pour les comparer.

Comme pour les traitements de texte, les commandes sont différenciées des touches de caractères par la méthode de l'appui simultané d'une touche fonction ; par exemple : "CONTROL" et "P" pour les commande d'impression ou "POMME OUVERTE" et "?" pour l'appel au menu d'aide. L'utilisateur aura encore à apprendre le code de ces fonctions ou à faire appel au résumé des commandes ; à cet égard, un progiciel utilisant la souris est toujours d'un maniement plus naturel.

Pour simplifier la lecture, il est commode de pouvoir taper effectivement des formules telles que : "total = recette 1 + recette 2" plutôt que sa traduction en termes de coordonnées du tableur, ce qui donne par exemple : L2C4 + L8C17 = L9C17.

MULTIPLAN

MULTIPLAN est à l'heure actuelle le tableur le plus répandu sur le marché. Pour les possesseurs de l'Apple //c, il a l'inconvénient de ne pas proposer l'option souris.

Présentation : Avec ses 128 Ko de mémoire centrale, l'Apple //c permet d'utiliser MULTIPLAN sans problème — rappelons que les 64 Ko de l'Apple //e étaient insuffisants et qu'il fallait acheter une extension de mémoire de 64 Ko.

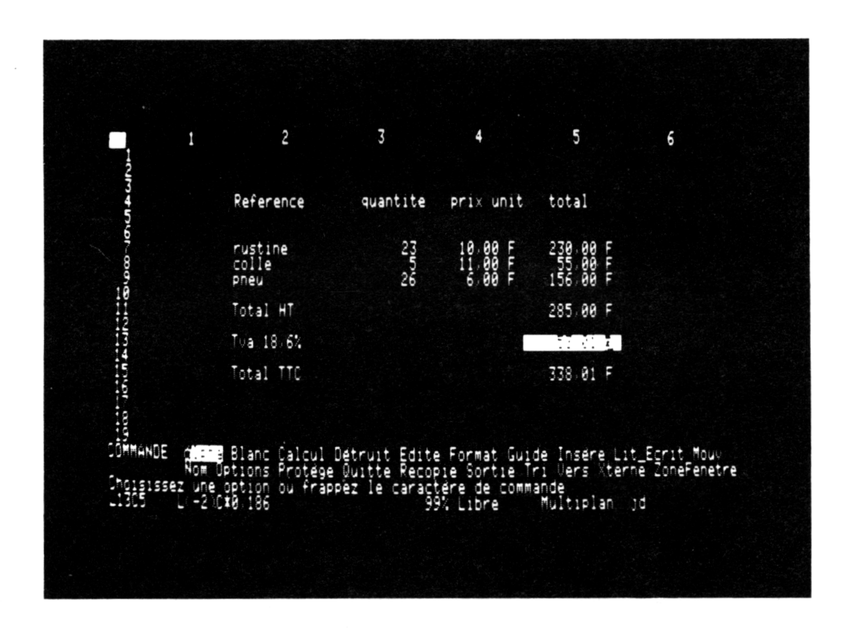
Le progiciel est enregistré sur deux disquettes :
— une disquette de chargement (que l'on nomme aussi "boot strap") ;
— la disquette du programme.

La disquette contenant le programme est lue régulièrement et s'use donc plus vite que la disquette de chargement qui ne sert qu'une fois par utilisation ; la société MICROSOFT qui a développé MULTIPLAN a prévu que l'utilisateur fasse plusieurs copies de la disquette programme tandis que la disquette de chargement ne tolère normalement qu'une unique recopie.

Lorsque les deux disquettes ont été chargées successivement dans la mémoire centrale, l'écran apparaît partagé en deux zones :

• une zone de travail qui est une fenêtre de la grande grille du tableau ;

• une zone de commandes qui fait référence aux options de MULTIPLAN (c'est un menu disposé sous l'espace de travail).



Le repérage d'une case s'effectue en lignes et colonnes. Par exemple : la case de la ligne n° 2 et de la colonne n° 5 se tape L2C5.

Dans sa totalité, le tableau possède 255 colonnes et 63 lignes. L'écran de travail affiche 10 colonnes et 18 lignes sur l'Apple //c qui dispose d'une option lui permettant d'écrire 80 caractères sur une ligne d'écran.

A priori, une case peut contenir 10 caractères mais des possibilités sont offertes à l'utilisateur pour redéfinir le format des cases. Le libellé d'un titre est souvent trop long pour apparaître complètement dans une case type mais il est tout de même intéressant de le lire en entier : l'option "FORMAT" du menu permet d'agrandir les cases et de cadrer les titres dans leur logement.

Deux possibilités existent pour écrire dans une case :

- l'option "ALPHA" pour alphanumérique si on entre des caractères quelconques (lettres, chiffres, symboles) ;

- l'option "CALCUL" pour taper des données numériques uniquement.

Les formules mathématiques sont entrées sous l'option "CALCUL" et ont la forme des coordonnées de cases comme par exemple :

(L11C2 + L6C5)/(L4C2 – (L4C2*L20C2) – L18C2)

Comme pour tous les progiciels, la zone travail de l'utilisateur est un fichier pour le micro-ordinateur. Ce fichier réside en mémoire centrale et peut être transféré sur une mémoire secondaire (une disquette) à volonté. Sur MULTIPLAN, l'option principale s'appelle "LIT-ECRIT". Elle ouvre sur un menu secondaire où les choix recouvrent les commandes habituelles d'un système d'exploitation de disquettes : charge, sauvegarde, efface écran, détruit option, renomme.

Ces programmes systèmes sont d'ailleurs compatibles avec le DOS 3.3, ce qui signifie que l'on peut sauver un fichier créé par MULTIPLAN sur n'importe quelle disquette formatée sous PRODOS.

LES GESTIONNAIRES DE FICHIERS (base de données)

Il n'est pas inutile de préciser que le terme de ''base de données'' a un sens très précis pour les spécialistes travaillant sur de gros systèmes informatiques, et un autre sens dans le cadre de la micro-informatique.

Schématiquement, disons qu'une banque de données est un ensemble d'informations se rapportant à un sujet précis et qu'une base de données ou plus précisément un logiciel de base de données est un ensemble de programmes dont la fonction est de gérer informatiquement ces informations. Les informations se trouvent stockées dans des fichiers et la difficulté consiste à rechercher et à trier les données éparpillées dans différents fichiers.

Exemple : l'annuaire électronique vous met en relation avec la base de données contenant les informations sur les abonnés du téléphone.

Dans le cadre de la micro-informatique, le terme de progiciel de base de données se confond souvent (pour des raisons de publicité commerciale) avec celui de gestion de fichiers. Mais la différence est importante et immédiate : un logiciel de gestion de fichiers ne peut traiter à la fois qu'un fichier principal unique alors qu'un logiciel de gestion de base de données est capable de traiter des informations archivées sur plusieurs fichiers de plusieurs disquettes.

Il est évident que les seconds sont bien plus compliqués que les premiers, demandent une configuration plus importante — surtout en capacité de mémoire centrale — et sont d'une utilisation plus délicate. Quelques exemples vous montreront les limites des gestionnaires de fichiers et la puissance des gestionnaires de base de données.

LES LOGICIELS DE GESTION DE FICHIERS

Les logiciels de gestion de fichiers ne sont capables de gérer qu'un fichier unique : celui de l'utilisateur. L'élément de base d'un tel fichier est l'enregistrement appelé aussi fiche ; chaque enregistrement est composé lui-même de champs.

Exemple : dans un fichier d'adresses, l'enregistrement contient une adresse, composée des champs nom, prénom, rue, code postal...

Le fichier créé par l'utilisateur est ainsi constitué d'une succession de fiches "à plat".

Champ Nom	Champ adresse	Champ INSEE
ROELLON Sylvie	116 Bd St Bermain Paris 6	2600575795225
PONT-LAPILE Serge	17 Bd Richard Leblanc Paris 11	1500691231512
PRODET Thierry	5 rue Epée de Fer Paris 5	1590872511014
PREBOIS Laurent	2 Impasse Galliéni Paris 1	1171282505214

Les programmes de gestion de fichiers offrent toujours les options minimum suivantes :

• définition de la structure du fichier, c'est-à-dire de la longueur de chaque enregistrement et du type d'information que va recevoir chaque champ ;

• le travail sur les enregistrements comme, par exemple, la création de nouvelles fiches ou la modification de celles déjà existantes ;

• l'édition sur écran ou sur papier du contenu du fichier.

Pour être efficaces, les gestionnaires de fichiers doivent disposer d'options de traitement des fiches. L'option "tri" permet d'éditer les fiches suivant des critères définis par l'utilisateur.

Par exemple, on peut demander au programme d'imprimer les fiches dans l'ordre alphabétique des noms ou dans l'ordre croissant des numéros.

Les différentes possibilités de tri, y compris la combinaison de plusieurs critères, donnent sa valeur au progiciel de gestion de fichiers. L'utilisateur devra calculer la taille de son fichier principal en fonction du nombre de tris qu'il désire mettre en œuvre : en effet, ces deux options influent sur la place mémoire nécessaire. Avec 128 Ko de mémoire centrale, l'Apple //c offre une configuration favorable à tous les logiciels existant sous DOS 3.3.

Le langage d'interrogation caractérise également la puissance du logiciel. Ainsi, si l'on peut toujours poser l'interrogation suivante : "éditer toutes les fiches dont les références sont comprises entre 20 et 43", il n'est pas évident que le programme soit capable de traiter le fichier suivant plusieurs critères comme : "sortir toutes les fiches commençant par N ou M pour lesquelles les références sont comprises entre 20 et 43 et dont le prix est supérieur à 1 900". Pour une utilisation souple, des interrogations portant sur plusieurs critères doivent se définir facilement et être lisibles.

VISIFILE

VISIFILE est un système de gestion de fichiers offrant la possibilité d'impression et de tri suivant plusieurs critères. Il se présente sous la forme de deux disquettes qui contiennent le programme. Suivant la configuration dont vous disposez, et plus précisément du nombre de lecteurs de disquettes, vous aurez à "configurer le programme" lors de sa première utilisation (c'est-à-dire à préciser un certain nombre de paramètres).

Dès que le programme est chargé dans la mémoire de l'Apple //c, un menu apparaît sur l'écran. Il propose à l'utilisateur toutes les options classiques : éditer (ajout, suppression, insertion de nouvelles données) ; effectuer des copies de sauvegarde ; définir plusieurs formats d'écran ou d'impression; changer la structure d'un fichier (sans perdre les informations contenues).

Les tris sont proposés dans une option spéciale SORT et les commandes du DOS sont appellées ici : “Menu de fonctions spéciales”.

VISIFILE adresse systématiquement à l’utilisateur la page du menu principal, et à chaque sélection il affiche l’ensemble des commandes disponibles. L’utilisateur choisit son option en déplaçant le curseur sur la fonction désirée et valide en appuyant sur “RETURN”.

Fichier : CELEB   Revue/Ajout/Modif.   Escape : Menu Principal

Sélection : toutes les fiches

Pseudonyme	Nom réel	Signe	Né en	Profession
Crawford, Joan	Le Sueur, Lucille	Bélier	1908	cinéma
Dylan, Bob	Zimmerman, Robert	Gémaux	1941	chanson
Ford, John	O’Feeney, Sean	Verseau	1895	cinéma
John, Elton	Reginald, Dwight	Bélier	1947	chanson
Montand, Yves	Livi, Ivo	Balance	1921	cinéma
Rooney, Mickey	Yule, John Jr.	Balance	1920	cinéma
Holden, William	Beedle, William Franklin	Bélier	1918	cinéma
Harlow, Jean	Carpenter, Harlean	Poisson	1911	cinéma
Douglas, Kirk	Demsky, Issur	Sagittaire	1916	cinéma
Hudson, Rock	Fitzgerald, Roy Scherer	Scorpion	1925	cinéma
Piaf, Edith	Gassion, Edith	Sagittaire	1915	chanson

Tapez un texte ou options   Ligne 3 Colonne 2   ó-? pour Aide

Quelques caractéristiques de VISIFILE : la longueur des articles est limitée à 232 caractères répartis sur 24 champs au maximum. Cette ordre de grandeur vous laissera penser que VISIFILE est avant tout un “petit système de gestion de fichiers”.

Une seconde remarque fixe aussi ses limites en tant que progiciel professionnel : il a été écrit en langage BASIC et n’est donc pas très rapide d’exécution (les progiciels rapides sont écrits en assembleur en général).

Autres progiciels de gestion de fichiers : PFS FILE, QUICK FILE.

LES SYSTÈMES DE GESTION DE BASES DE DONNÉES

Les systèmes de gestion de base de données ont été écrits dans un état d'esprit bien plus ambitieux et conviennent parfaitement aux applications professionnelles. Leur prix, la configuration requise et l'investissement à fournir garantissent d'ailleurs que ces progiciels ne seront pas utilisés par des non-spécialistes.

Prenons par exemple le cas d'un fichier utilisateur double par opposition au fichier utilisateur unique du système de gestion de fichiers. La puissance d'un système de gestion de base de données apparaît avec la possibilité de définir des liens logiques entre les deux fichiers : on associe des informations contenues dans un des fichiers à celles contenues dans l'autre.

Il faut remarquer qu'un tel progiciel ne se contente pas de manipuler plus d'informations deux fichiers au lieu d'un ; il sait relier une fiche de l'un à plusieurs fiches de l'autre.

Fichier fournisseur	
Nom	Adresse
Ets Dupuis	Rue Albert I Courbevoie
Ets Gaudin	Rue du départ Orsay

Liens

Fichier produit		
Nom	Référence	Quantité
Bureau	17342	17
Chaise	07345	56
Lampe	97855	6
Lit	24376	12
Drap	54897	50

Exemple type : gestion de stock ; le premier fichier est un fichier fournisseur (noms, adresses...) ; le second fichier est un fichier produits (références, description, quantité, prix...).

Le gestionnaire de base de données permet d'associer un (ou plusieurs) article à un (ou plusieurs) fournisseur.

Les progiciels décrits ici permettent de résoudre des problèmes relativement complexes. On comprend que les utilisateurs de gestionnaires de fichiers et de gestionnaires de base de données ne soient pas vraiment les mêmes personnes. Seule la publicité a tendance à confondre les deux, au profit des premiers.

Le premier progiciel de base de données apparu sur le marché s'appelle DBASE II. Il tourne sous le système d'exploitation CP/M et n'est donc pas implantable sur l'Apple //c. CX Multigestion en revanche est un progiciel tournant sous DOS 3.3 et donc parfaitement accepté par l'Apple //c.

Les logiciels de base de données sont riches mais compliqués. Il est difficile d'en décrire ici la mise en marche comme on l'a fait pour un traitement de texte. Pour l'apprentissage, une disquette d'initiation est généralement fournie avec la documentation.

Il faut quand même s'attendre à passer quelques heures à se familiariser avec les différents "morceaux" du progiciel :

— la définition du schéma de base (on y accède par l'ensemble des commandes de définition) ;

— les commandes de manipulation des données (elles permettent les ajouts, les suppressions et les modifications) ;

— les commandes d'interrogation de la base (ou comment récupérer les informations stockées avec tant de difficulté !).

De tels logiciels sont encore peu répandus sur micro-ordinateurs car ils exigent une place mémoire importante. Cependant, les progrès effectués depuis ces dernières années et la demande professionnelle de plus en plus importante nous laissent penser que leur nombre devrait augmenter rapidement dans un futur proche.

LES LOGICIELS INTÉGRÉS

Devant les besoins croissants de la micro-informatique professionnelle, les concepteurs de logiciels ont créé depuis peu les logiciels intégrés. Cette "nouvelle race" de logiciels correspond à un besoin précis : comment faire parvenir une lettre personnalisée qui serait fabriquée à partir de plusieurs fichiers gérés par ailleurs et qui inclurait des tableaux actualisés eux aussi par un programme. Une première tentative de réponse a été de faire lire les fichiers créés par un progiciel comme données pour un autre progiciel (par exemple : VISIFILE et VISICALC peuvent communiquer).

Un logiciel intégré est un programme qui comprend, sous une même organisation, plusieurs fonctions classiques d'un progiciel. On trouve ainsi réunis le plus souvent : un traitement de texte, un tableur, un système de gestion de fichiers.

Quelquefois, ces trois options de base sont accompagnées de logiciels graphiques ou d'autres applications particulières.

L'avantage des progiciels intégrés est évident : ils sont créés pour faire communiquer des fichiers d'une application à une autre, et inclure un morceau de tableau dans un texte n'est plus l'exception mais la règle.

Les logiciels intégrés utilisent au maximum le système du fenêtrage. L'écran de travail de l'utilisateur est découpé en plusieurs zones ; dans chacune d'elles, on peut avoir accès à une option. L'utilisateur appelle successivement chaque option s'il désire travailler sur tout l'écran ou garde le partage en permanence.

De par sa conception, un logiciel intégré autorise l'utilisateur à user des mêmes fonctions dans chacune des fenêtres. C'est ainsi que supprimer un mot dans un programme de traitement de texte et supprimer le contenu d'une case d'un tableur suivent les mêmes commandes. C'est évidemment simplificateur ! Mais le programme devient magique (!) lorsqu'on copie une portion de tableau dans la fenêtre du traitement de texte, exactement comme s'il s'agissait d'un texte classique.

JANE

JANE regroupe les trois options classiques d'un logiciel intégré, et vous n'aurez pas de difficultés à identifier le traitement de texte, le tableur et le système de gestion de fichiers parmi les trois noms suivants : JANEWRITE, JANECALC et JANELIST.

JANE ne consomme que 64 Ko de mémoire vive, ce qui est un véritable exploit car les logiciels intégrés ont la réputation d'être gourmands en place mémoire (la création de fenêtre est une option nécessitant une zone mémoire importante).

Pour terminer la description de la configuration, signalons que JANE se contente d'une unité de disquette mais qu'il exige une souris.

Comme EPISTOLE, le logiciel intégré JANE utilise des séries d'icônes placées au-dessus et au-dessous de l'écran de travail.



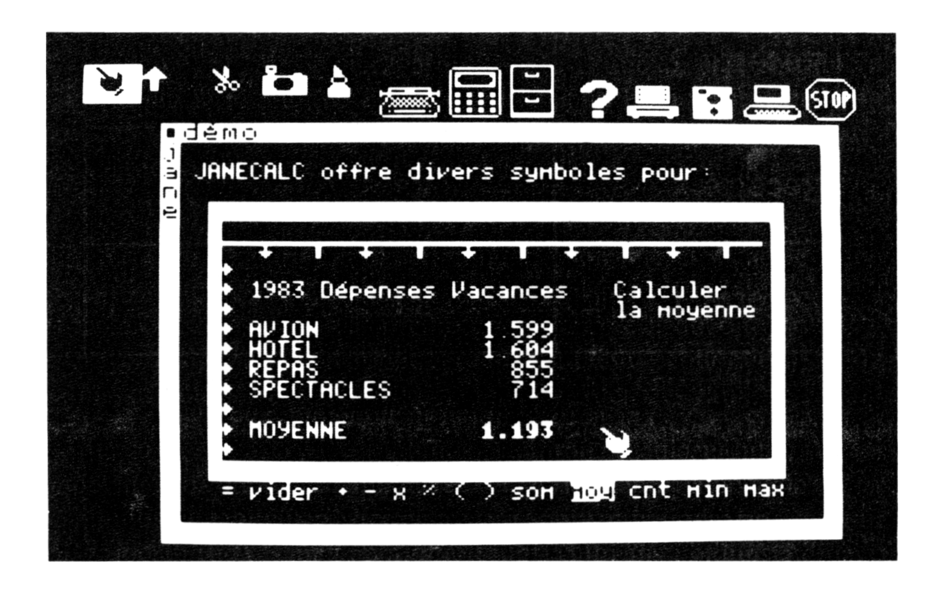
Logiciel fourni par Galilée Informatique

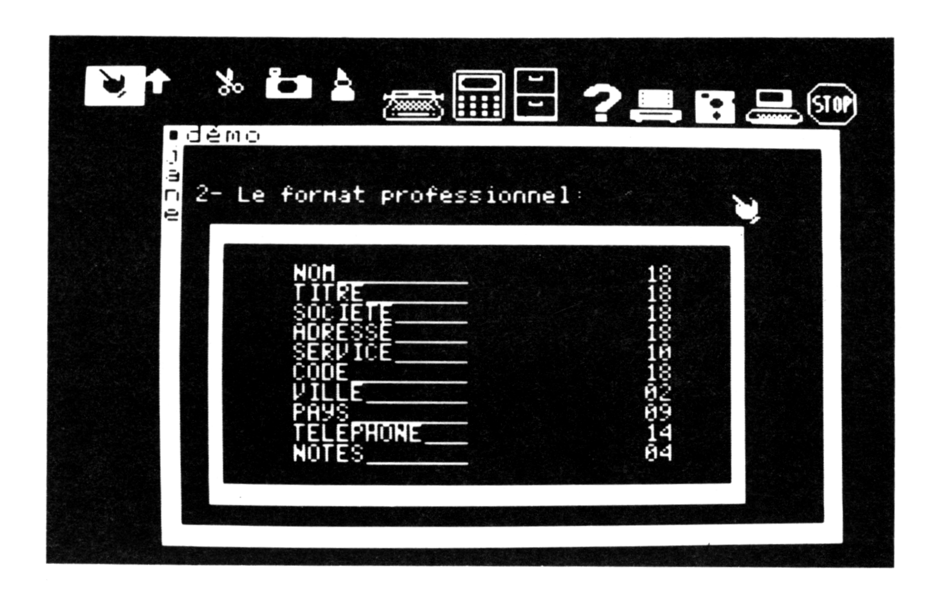
Cette formule raccourcit d'autant la durée d'apprentissage, et de toute façon, une disquette d'aide est fournie avec le progiciel. Elle contient les renseignements utiles en cas d'hésitation (et sous forme de dessins animés !).

Les commandes à utiliser pour transférer des informations du tableur ou du gestionnaire de fichiers vers le traitement de texte sont simples et parlantes : on "photographie" le passage à recopier (icône : appareil photo) puis on le "recolle" (icône : pot de colle) à l'endroit choisi. La manœuvre se résume en une suite de déplacements du curseur et de validations ; tout est commandé par la souris.

Le nombre de fenêtres affichables simultanément à l'écran donne une idée de la puissance d'un logiciel intégré. JANE est un "petit" logiciel intégré puisqu'il ne peut faire apparaître que quatre fenêtres au maximum. A titre de comparaison, les gros logiciels intégrés (comparables à LOTUS 1-2-3 par exemple) acceptent plus de dix fenêtres. Mais l'Apple //c a 128 Ko de mémoire centrale et de tels logiciels en exigent bien plus. Cette remarque nous rappelle dans quelle gamme de micro-ordinateurs professionnels se trouve le petit compact d'Apple.

Les quatre fenêtres de JANE offrent quatre dossiers "empilés" à l'utilisateur. Il peut les déplacer, les ranger, les faire surgir, agrandir, réduire, etc. A l'intérieur de chaque fenêtre, il dispose de l'ensemble des commandes spécifiques de l'option affichée : traitement de texte, gestion de fichiers ou tableur.





Les caractéristiques de ces trois options ont été étudiées en détail dans les paragraphes précédents. Notons quand même quelques traits particuliers :

L'écran de travail JANEWRITE correspond exactement à l'écran de visualisation (chez EPISTOLE ou APPLE WRITER), c'est-à-dire que les caractères gras ou le soulignement apparaissent à l'écran sans que l'utilisateur ait besoin de taper des symboles spéciaux ; de même, la mise en page se fait sur l'écran de travail devant l'auteur et non pas juste avant l'impression. Le traitement de texte permet d'écrire des lignes de 80 caractères mais seule une fenêtre de 64 caractères sur 12 lignes apparaît.

Nota : le tréma et l'accent circonflexe se placent sur la lettre et non pas à côté !

JANELIST est un gestionnaire de fichiers proposant d'emblée deux types de fiches pré-établies par défaut (nom + adresse + ... et nom + titre + société + ...). Il permet d'écrire des lignes de 250 caractères se répartissant en dix rubriques de 25 caractères. A titre indicatif, les disquettes de l'Apple //c pourront stocker environ 500 fiches ainsi définies.

Le tableur JANECALC a la particularité de laisser l'utilisateur placer chiffres et titres où bon lui semble (centrés ou non, sur deux colonnes...), il suffit de sélectionner la bonne icône. De même, certains calculs en lignes ou en colonnes s'effectuent en promenant le curseur (toujours après avoir sélectionné l'option de calcul adéquate).

Ces facilités de commande se "paient" sur la taille du tableau (24 lignes d'une vingtaine de colonnes seulement. L'explication tient toujours à la capacité mémoire : JANE réclame 64 Ko. Les possesseurs de l'Apple //c espèrent donc une version augmentée pour leurs 128 Ko, qui garderait la même souplesse d'utilisation.

Un autre logiciel intégré est disponible sur l'Apple //c. Il est développé par la société Apple elle-même et s'appelle Apple Works.

Il a bien sûr toutes les fonctions du logiciel intégré décrites ci-dessus, mais nous ne nous étendrons pas sur son fonctionnement et ses possibilités. En effet, l'Apple //c est livré avec une série de disquettes de démonstration et l'une d'elles contient une présentation d' Apple Works.

LES AUTRES APPLICATIONS LOGICIELLES DE L'APPLE //c

Les amateurs de jeux n'auront pas besoin de ce livre sur l'Apple //c pour découvrir les possibilités de leur machine. La compatibilité ascendante DOS 3.3-PRODOS permet de charger toute disquette de jeux prévue pour l'Apple //e, mais pas pour l'Apple //+. Il faut vérifier jeu par jeu...

Le graphisme

A mi-chemin entre les applications professionnelles et la passion du programmeur fou, on trouve le désir de "faire des dessins" avec son micro-ordinateur.

Quelques logiciels de graphisme permettent d'aider le dessinateur à réaliser ses premières œuvres. Le problème posé est relativement ardu : montrer une forme à trois dimensions grâce à un effet de perspective ou animer un petit "lutin" (traduction de l'anglais *sprite*) à l'aide de mouvements non simplistes relève rapidement de la CAO (Conception assistée par ordinateur) ou de la programmation en assembleur.

Il existe des livres entiers de programmes donnant toutes les clés pour dessiner avec l'Apple //e. La traduction pour l'Apple //c est automatique.

A titre d'exemple, citons deux progiciels :

— Apple Word qui permet l'animation graphique en trois dimensions,
— Graphor qui n'autorise que la création graphique en deux dimensions.

Ces exemples traduisent des conceptions et ambitions très différentes.

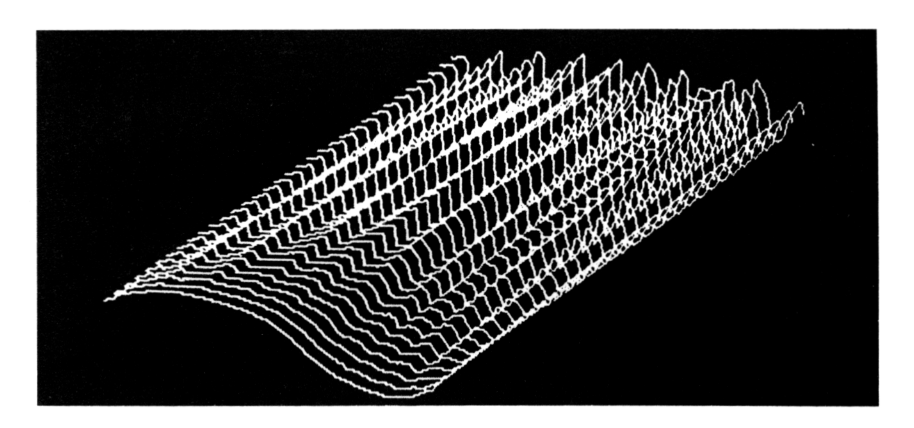
Le principal intérêt de ce logiciel réside dans sa simplicité. Notons qu'il permet de sauver sur une disquette les dessins obtenus à l'écran ; un programme BASIC pourra les rappeler à l'aide de deux instructions.

AppleWord a une ambition plus grande : faire des dessins en trois dimensions et les animer. Il est clair qu'une bonne connaissance des notions de coordonnées relatives et absolues ainsi que des repères cartésiens munis d'une échelle permet de surmonter les obstacles du langage.

La qualité des dessins obtenus est dépendante de la résolution de l'écran, c'est-à-dire du nombre de pixels adressables à l'écran. L'Apple //c a une résolution maximum de 280 par 192. La limitation vient toujours d'un problème de place mémoire : la page graphique a une correspondance dans la RAM réservée au système et sa taille a été prédéterminée à la construction du micro-ordinateur.

Graphor permet de tracer des lignes régulières et des formes de géométrie simple (triangle, carré, cercle...). On commande pour cela le déplacement du curseur suivant huit directions, à la manière des points cardinaux : N, S, E, O, NE, NO, SE, SO.

Le concepteur doit donc "calculer mathématiquement" son dessin et le programmer ensuite. Exemple de dessin obtenu :



Si la saisie des coordonnées peut paraître longue, la satisfaction est grande de voir son dessin apparaître. Les commandes pour le faire tourner sur lui-même et bouger sur l'écran sont très simples.

Remarque pour les amateurs : les faces cachées ne sont pas reconnues comme telles.

Il faut d'ailleurs, d'une façon générale, apprendre à apprécier les qualités réelles des logiciels graphiques. On a tendance à trop exiger : des traits fins, des lignes parfaites, une bonne évaluation du volume. Quand toutes ces qualités sont réunies, on obtient un véritable outil d'aide à la conception par ordinateur. Il faut alors des écrans graphiques de 600 × 600 pixels — voire 1 000 par 1 000 — et le logiciel obtenu coûte bien plus cher. La saisie des coordonnées se fait sur un périphérique adapté (une table traçante) qui soulage l'utilisateur de la frappe des nombreuses lignes de programme.

Cette application relève bien plus du bureau d'étude que de la micro-informatique personnelle. Mais, comme certaines applications peuvent demander de tels développements, l'Apple //c peut être configuré pour répondre à ces besoins — à son échelle, bien entendu !

UN LOGICIEL A PART : THINKTANK

La publicité américaine présente THINKTANK comme : ''*The first Idea Processor*''. Et la première bonne question à se poser est : ''*What is an Idea Processor ?*'' : et accessoirement : ''Quelle traduction française donne-t-on d'*Idea Processor* ?''

Le principe de THINKTANK

THINKTANK sert à ordonner des idées, ce qui se traduit avec un micro-ordinateur et un écran par du travail sur des paragraphes, donc sur du texte. On trouve d'ailleurs dans THINKTANK les commandes classiques d'un traitement de texte, c'est-à-dire la suppression de lettre, de mot, l'insertion, etc.

L'originalité de THINKTANK réside dans la façon d'ordonner le texte. Avec le traitement de texte, l'auteur écrit un article au kilomètre, ligne après ligne, paragraphe après paragraphe et page après page. Il a le plan dans sa tête ou noté à part. THINKTANK permet (et oblige) son utilisateur à définir son plan avant d'écrire. Il le force à réfléchir au nombre de rubriques à inclure et à leur agencement avant de fournir la page de texte dans laquelle on peut taper.

Pour les informaticiens un peu avertis, on peut dire que THINKTANK repose sur la notion de ''programmation structurée''. Il n'est d'ailleurs pas surprenant qu'il fonctionne sous le système d'exploitation PASCAL UCSD et soit écrit à la fois en PASCAL et en assembleur (rappelons que PASCAL a introduit la notion de programmation structurée).

Concrètement, THINKTANK permet d'avoir une vision synoptique ou au contraire détaillée de son ouvrage : il agit comme une loupe. L'utilisateur peut donc visualiser successivement les têtes de chapitres de son ouvrage, puis les têtes de chapitres accompagnées des sous-chapitres, etc... jusqu'au texte effectif.

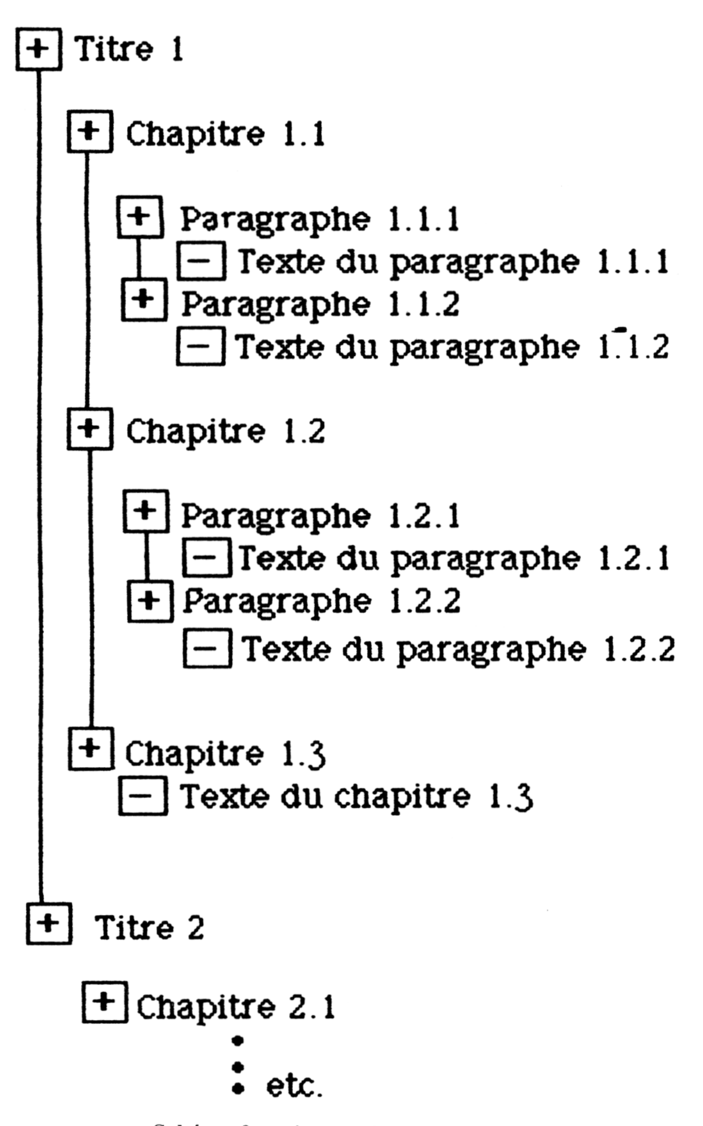


Schéma fonctionnel de THINKTANK

• Les rubriques de même niveau (titre, chapitre, paragraphe) sont reliées par un trait.

• Les rubriques précédées d'un + indiquent qu'elles appellent des sous-rubriques.

• Les rubriques précédées d'un — indiquent le début d'un texte.

THINKTANK est disponible pour toute la gamme Apple sous différentes versions mais l'Apple //c n'a pas (encore) de version personnalisée ; nous commenterons donc la version développée pour le //+ et le //e, qui nécessite 64 Ko de mémoire vive. Elle n'utilise pas à fond les possibilités des 128 Ko et de la souris du //c, ni même vraiment les touches fonction "POMME OUVERTE" et "POMME FERMEE" qui n'existent pas sur le //+.

THINKTANK se présente sous la forme de deux disquettes :
— une "disquette programme",
— une "disquette données".

Il demande donc deux unités de disquettes car le programme n'est pas présent complètement en mémoire centrale. Seule les "portions de codes" nécessaires à un moment donné sont chargées, et les échanges entre les unités de disquettes et le micro-ordinateur sont réguliers.

L'utilisateur se trouvera ici confronté à un problème nouveau : la pénalisation due au temps d'accès sur disquettes (un ordre de grandeur d'échange : 5 à 10 secondes). La lenteur d'exécution n'est pas causée par la structure ou le langage de développement (comme VISIFILE écrit en BASIC par exemple) mais par la trop grande place mémoire qu'il exige et qui l'empêche d'être résident.

Une fois lancé, THINKTANK propose un écran de travail où les quatre dernières lignes sont réservées aux menus, commandes, renseignements et messages d'erreurs.

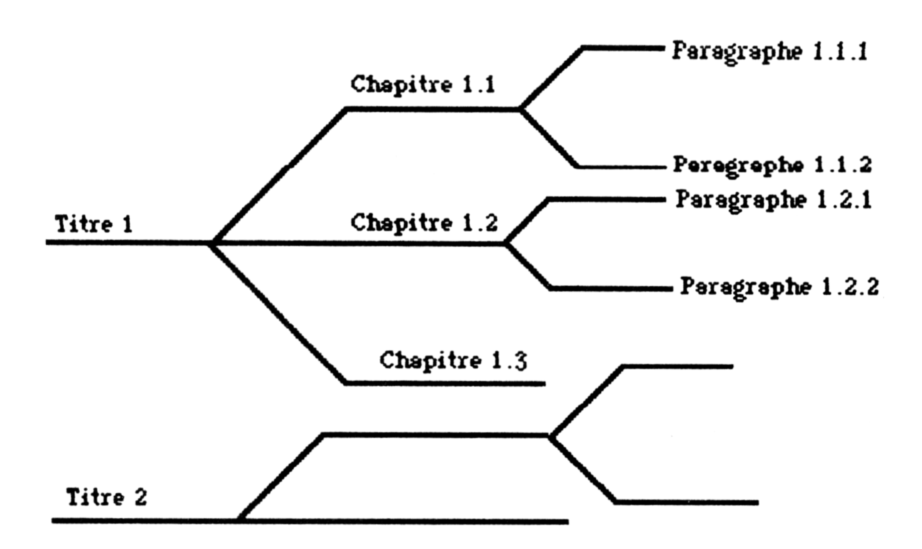
Le dialogue entre l'utilisateur et le progiciel utilise un code dont les symboles sont le "plus" et le "moins".

Une ligne commençant par un "plus" signifie que l'on a affaire à un titre de rubrique qui peut ouvrir sur plusieurs sous-rubriques.

Ces sous-rubriques ont des libellés et peuvent aussi commencer par un "plus" signifiant qu'elles-mêmes possèdent des sous-chapitres.

Une ligne commençant par un "moins" signifie que derrière le titre s'ouvre une page de texte par opposition à une page de sous-chapitre. La page texte peut contenir 2 048 caractères et s'appelle un paragraphe. La structure générale obtenue est celle d'un arbre.

Pour se repérer dans les différents niveaux, l'utilisateur dispose d'un "curseur-barre" qui met en inverse vidéo toute la ligne de titre sur laquelle il est positionné.



Les principales commandes portant sur les paragraphes permettent d'ouvrir ou de fermer des niveaux de titres. Il est possible de changer le niveau d'une ligne de titre et de déplacer des paragraphes entiers. Ces manœuvres demandent la frappe de quelques touches qui ont les initiales des commandes : elles sont rappellées dans les quatre lignes du bas par la touche "SLASH" (/) — dans la version anglaise. Au niveau du paragraphe, les commandes portant sur le texte sont équivalentes à celles d'un traitement de texte.

THINKTANK dispose aussi des commandes d'impression. Il faut néanmoins définir les formats de sortie et les paramètres d'impression : longueur des lignes, espacement, nombre de copies, mise en page, etc. Le programme vous propose également une numérotation automatique de vos lignes de titres sous la forme : 1., 1.1, 1.1.1, 1.1.2, 1.2, etc.

Ce progiciel possède également une fonction de tri basée sur le classement alphabétique des rubriques de même niveau.

THINKTANK n'a pas toutes les qualités d'un progiciel de traitement de texte mais encore quelques versions et il pourra réellement les concurrencer.

Il apportera alors une nouvelle façon d'organiser l'écriture grâce aux notions de titres et niveaux de chapitres.

Remarque : on peut préférer une écriture dans un fichier plutôt qu'une sortie à l'imprimante. THINKTANK permet d'écrire dans des fichiers PASCAL de type texte (rappelons que THINKTANK "tourne sous" PASCAL UCSD). Un autre programme pourra aller relire le texte et travailler dessus s'il le faut.

LES LANGAGES

PASCAL

PASCAL est un langage de programmation de haut niveau créé en 1970 par Niklaus Wirth de l'Institut de Technologie de Zurich.

Il fut conçu initialement pour l'enseignement de la programmation mais aujourd'hui, il est l'un des langages les plus répandus et les plus utilisés pour le développement.

Les programmes écrits en PASCAL sont très lisibles puisque décomposables en blocs logiques ou procédures. Contrairement au BASIC, les lignes ne sont pas numérotées.

C'est le langage type de la programmation structurée, et il est à l'origine de bien d'autres langages tels que Modulo 2 ou ADA.

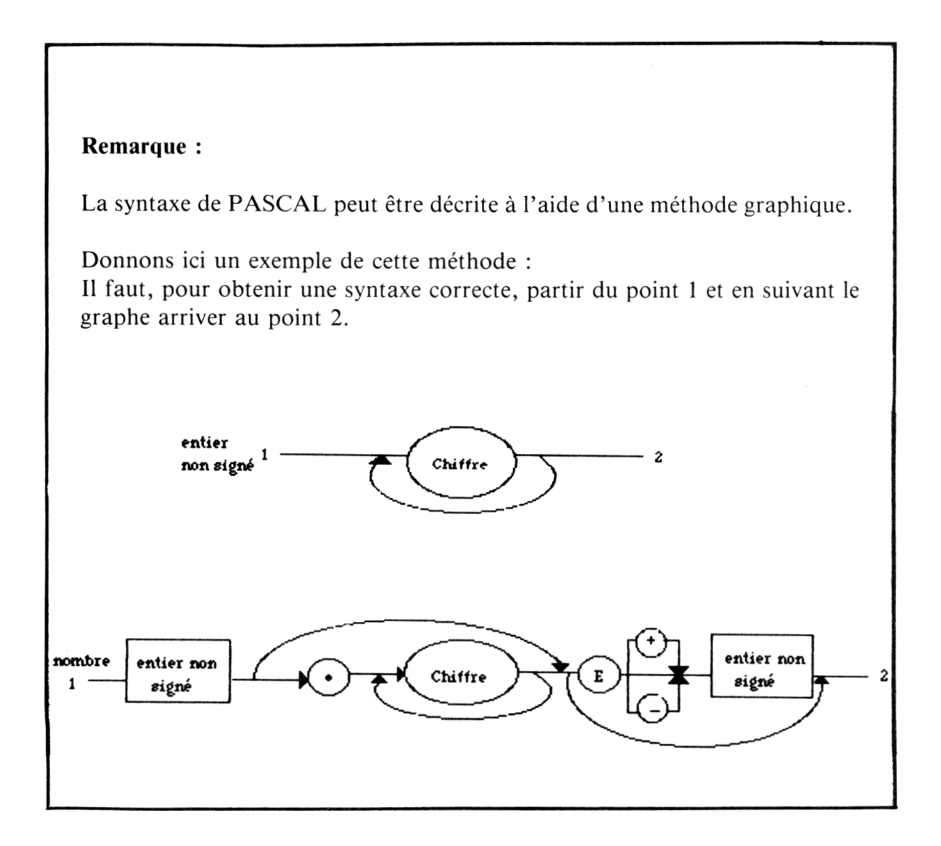
PASCAL diffère également de BASIC par le fait que toute variable utilisée dans un programme doit faire l'objet d'une déclaration précisant son type.

En plus des types classiques (entiers, réels, chaînes de caractères) PASCAL accepte des déclarations de type ayant la forme d'ensembles ou de collections d'objets de types différents et en nombre variable.

Remarque :

La syntaxe de PASCAL peut être décrite à l'aide d'une méthode graphique.

Donnons ici un exemple de cette méthode :
Il faut, pour obtenir une syntaxe correcte, partir du point 1 et en suivant le graphe arriver au point 2.



PASCAL, comme BASIC, est un langage descriptif (on doit prescrire la manière d'obtenir les résultats), et existe essentiellement en version *compilée*. Le langage acheté dans le commerce pour l'Apple //c se présente sous la forme de deux disquettes : l'une contient le compilateur et l'autre l'éditeur.

En effet, le PASCAL de l'Apple //c fonctionne sous son propre système d'exploitation qui remplace alors le système PRODOS.

On trouve donc sur ces disquettes le compilateur, le système d'exploitation UCSD ainsi que divers utilitaires tels qu'une bibliothèque de fonctions mathématiques et un programme appelé éditeur de liens, qui autorise la compilation séparée de divers programmes.

Notons que ces deux disquettes ne peuvent être chargées d'un seul bloc en mémoire centrale, leur capacité étant trop importante. Ceci ne pose pas de réel problème, le compilateur n'ayant pas besoin d'être résident en RAM au moment de l'écriture du programme et de son exécution, contrairement à un langage interprété.

Il est possible de programmer en PASCAL avec le seul drive intégré de l'Apple //c, quitte à jongler avec les deux disquettes. Il est donc recommandé d'utiliser deux drives pour une manipulation plus « confortable ».

BASIC

Le BASIC est un langage de programmation évolué répandu en micro-informatique. BASIC est une abréviation de Beginner's All Purpose Symbolic Instruction Code.

Il fut implémenté pour la première fois en 1965 par une équipe du Darmouth College.

C'est le premier langage destiné aux non-professionnels de la programmation. C'est un langage *interprété* qui permet la mise au point rapide de petits programmes.

Cependant, sa simplicité n'autorise pas l'écriture de logiciels importants à vocation purement professionnelle. En effet, il ne permet pas une programmation structurée ni la création de types de données complexes.

BASIC est maintenant disponible sur la quasi-totalité des micro-ordinateurs, mais dans des versions qui diffèrent notablement. Il est donc de faible portabilité d'une machine à l'autre et il conviendra d'effectuer un certain nombre de modifications pour exécuter des programmes BASIC sur un micro-ordinateur autre que celui pour lequel ils ont été écrits.

Le BASIC existe en version compilée et exécutée, et il est quasiment le seul langage descriptif possédant cette qualité. Il permet donc, en version exécutée, l'écriture de logiciels conventionnels qui sont très répandus en micro-informatique.

Il faut le dire bien clairement, le BASIC résident de l'Apple //c, identique à celui du //e, n'est pas un sujet de fierté. Certains ont parlé d'un BASIC archaïque... Il est vrai que de nombreux micro-ordinateurs aux possibilités techniques bien inférieures disposent d'un BASIC de meilleure qualité.

Mais, compatibilité oblige, sur ce terrain, les concepteur d'Apple //c ont fait preuve d'un conservatisme étonnant. Les amateurs de programmation regretteront donc l'absence des instructions ELSE, USING, REPEAT... UNTIL, WHILE... WEND, d'instructions graphiques autres que PLOT et l'absence totale d'instructions musicales. Ce qui ne signifie nullement qu'il soit impossible de gérer les graphiques et les sons en BASIC, mais les ''bidouilleurs'' restent condamnés à un usage intempestif des PEEK et autres POKE pour pallier les faiblesses des primitives. L'éditeur, légèrement amélioré par rapport au //e, n'est pas encore un modèle du genre, tant s'en faut.

Il reste donc à supposer, comme les constructeurs l'ont sans doute fait, que l'utilisateur moyen d'Apple //c s'intéresse à tout sauf à la programmation BASIC. Nous leur emboîtons donc le pas en passant à autre chose...

LOGO

« LOGO est un nom qui représente à la fois une philosophie d'éducation et une famille de langages pour ordinateurs, qui évoluent constamment, et en permettent l'évolution. L'univers d'apprentissage du LOGO développe et illustre le principe suivant : donnez aux gens le contrôle direct de ressources informatiques puissantes, et vous leur permettez d'établir un contact intime avec des concepts complexes tenant de la science, des mathématiques et de l'art de construire des modèles intellectuels. »

C'est ainsi qu'Harold Abelson introduit son livre *Le LOGO sur Apple* (Éditions Cedic/Nathan).

On ne parle pas de LOGO comme on parle de BASIC, la passion s'en mêle nécessairement. Car véritablement, LOGO n'est pas un langage comme les autres. Si l'on dit de BASIC qu'il est un langage *évolué*, il faudra dire de LOGO qu'il est un langage *évolutif*.

Développé en 1968 par S. Papert et A. Minski, au Massachusetts Institute of Technology, ce langage a dû attendre les années 80 et son implantation sur les micro-ordinateurs TI 99 et Apple // pour sortir de l'ombre et acquérir une réputation bien méritée dans le domaine éducatif.

Pourtant, avant de rentrer dans le détail, il est bon de souligner que les créateurs comme les utilisateurs de LOGO refusent l'amalgame systématique LOGO-éducation.

Ce n'est pas parce que les enfants parlent l'anglais qu'il faut conclure que l'anglais est une langue réservée aux enfants, fait observer Papert lui-même.

LOGO est un langage informatique puissant dérivé de LISP et dont les applications vont de la robotique à l'intelligence artificielle en passant par... l'apprentissage de la programmation.

En français, LOGO est disponible dans deux versions très semblables :

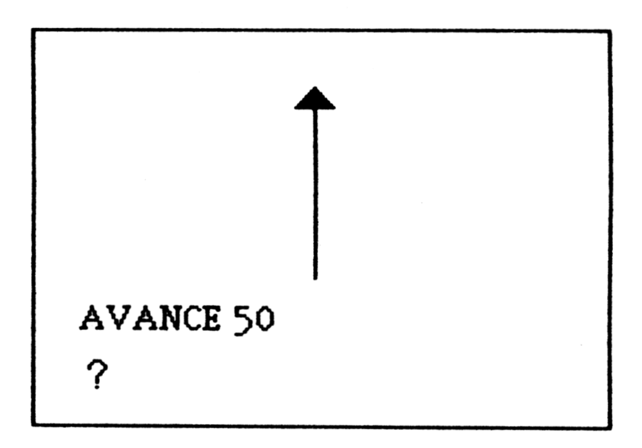
— Apple LOGO (Seedrin),
— Edi-LOGO (Ediciel).

Chacune se présente sous la forme d'une disquette et d'un manuel d'accompagnement qui permet de faire sans problème les premiers pas.

Il se trouve que les premiers pas en LOGO sont aussi ceux d'une tortue ou plus exactement d'un petit triangle isocèle pointant vers le haut de l'écran que tous les « logoïstes » appellent la tortue.

Ce triangle est présent au milieu de l'écran dès que l'interpréteur a été chargé. En bas, un point d'interrogation est une invitation claire au dialogue.

La programmation se fait en mode direct : vous donnez un ordre aussitôt exécuté. Puisqu'il s'agit d'une tortue, on peut avoir envie de la faire avancer et la première commande sera donc AVANCE 5Ø qui produira l'effet suivant :



Rien de plus simple, il suffit de savoir écrire des mots simples. Ensuite on donnera des ordres du style RECULE, DROITE, GAUCHE, REPETE, LEVE CRAYON, BAISSE CRAYON, VIDE ECRAN, etc.

C'est lors de ce premier contact que l'on peut mesurer l'intérêt pédagogique du langage LOGO : les enfants se fixent un projet, un dessin réalisable par la tortue,et ils tentent de donner à l'ordinateur les instructions adéquates.

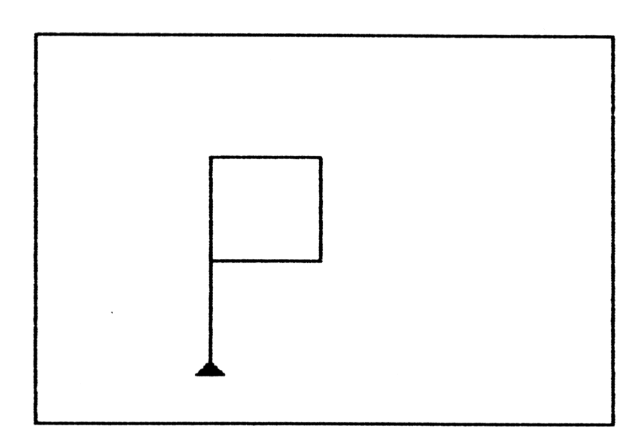
Erreurs, tâtonnements, retours en arrière font partie du processus d'auto-apprentissage. S. Papert, dans son fameux livre *Jaillissement de l'esprit* (Flammarion) décrit cela un peu pompeusement : « LOGO aide l'enfant à réfléchir sur sa propre pensée. »

Mais s'il fallait s'arrêter à ce point, tout cela ne mériterait pas tant de bruit. Le programmeur va pouvoir très simplement enrichir le vocabulaire en ajoutant de nouveaux mots que l'on appelle alors des *procédures*; les mots initiaux (AVANCE, RECULE, etc.) s'appellent des *primitives*.

Imaginons par exemple qu'une bouffée de patriotisme vous pousse à vouloir dessiner un drapeau. Rien de plus naturel que de créer une procédure, le mot DRAPEAU. Ce que vous ferez en tapant par exemple :

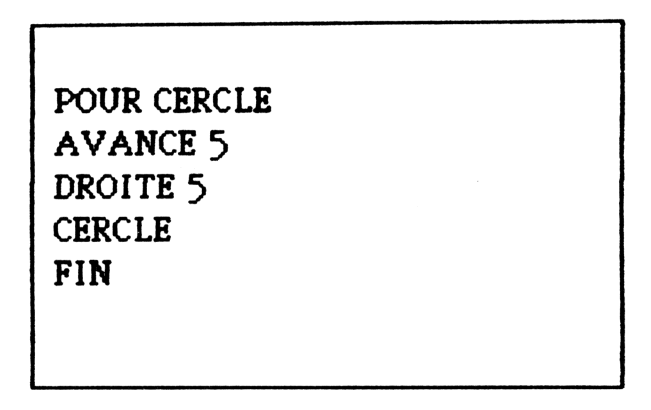
```
POUR DRAPEAU
AVANCE 50
REPETE 4 [AVANCE 30 DROITE 90]
RECULE 30
FIN
```

Pendant ce temps, il ne se passera rien au niveau de la tortue mais après le mot FIN, vous aurez droit au message : DRAPEAU EST DEFINI. Ce qui signifie : j'ai bien enregistré la définition de ce nouveau mot. Puisque le mot est connu, il suffit de l'employer en tapant DRAPEAU, ce qui a l'effet suivant :



DRAPEAU est devenu un mot que vous pouvez utiliser en toutes circonstances, à l'instar de AVANCE ou RECULE.

La possibilité de créer des procédures n'est pas le dernier avantage de LOGO : une procédure peut s'appeler elle-même, on dit que le langage est *récursif*. On pourra méditer longuement sur cette définition récursive du cercle en LOGO, si proche de l'expérience : pour faire un cercle, il faut avancer un peu, tourner un peu et recommencer, c'est-à-dire faire un cercle.



Lorsque vous aurez ainsi créé des procédures dont vous retirerez une légitime fierté, vous pourrez les ranger aux côtés des primitives en tapant SAUVE DRAPEAU ou SAUVE CERCLE qui enverront se ranger les mots DRAPEAU et CERCLE aux côtés des autres.

Votre disquette LOGO n'est alors plus la même que celle du voisin.

Dernier avantage et non des moindres : LOGO emprunte à LISP la puissante notion de liste qui permet de traiter très simplement les séquences de mots (l'équivalent des chaînes de caractères du BASIC).

Grâce aux procédures, à la récursivité et aux listes, LOGO offre aux adultes l'occasion d'une aventure intellectuelle d'une grande qualité. Grâce à la tortue, il offre une possibilité d'une expérience semblable aux enfants les plus jeunes.

La présence simultanée d'enfants et d'Apple //c dans une maison est presque une obligation pour y introduire un troisième acteur : LOGO.

LES PERIPHERIQUES

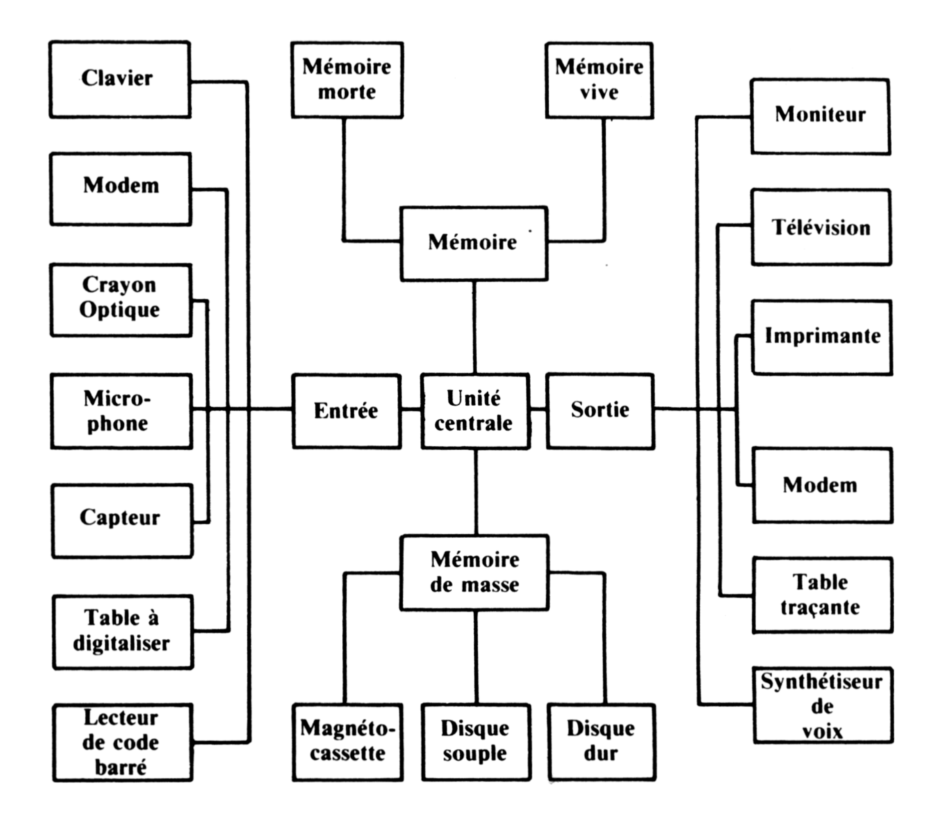
Un micro-ordinateur isolé n'a pas grand intérêt. Il lui est nécessaire de communiquer avec l'extérieur pour aller puiser des renseignements et restituer des résultats. Un périphérique est un matériel qui remplit cette fonction.

On distingue plusieurs catégories de périphériques suivant le type de services qu'ils rendent au micro-ordinateur :

— les périphériques de stockage dont un exemple est le lecteur de disquettes intégré,

— les périphériques d'entrée/sortie qui permettent à l'utilisateur lui-même de dialoguer avec sa machine. Comme périphérique d'entrée/sortie, l'Apple //c ne dispose à la livraison que de son clavier.

L'achat de nouveaux matériels pour compléter votre installation et arriver ainsi à « la bonne configuration » est commandé par les applications que vous envisagez.



LES TRANSMISSIONS

Les micro-ordinateurs, comme les gros ordinateurs, traitent des masses d'information si importantes que les chiffres donnés pour exemples ne sont pas parlants. En revanche, il apparaît clairement que l'un des problèmes clés de la micro-informatique va consister à faire transiter correctement ces grandes quantités d'information.

Quand les échanges se produisent à l'intérieur du micro-ordinateur lui-même, on peut dire que la difficulté des transmissions échappe à l'usager. On dit que le mode de transmission des octets en provenance du microprocesseur et à destination de la mémoire centrale est transparent à l'utilisateur; c'est du ressort des concepteurs de la machine.

Il n'en est pas exactement de même pour les échanges entre le micro-ordinateur et les différents appareils qui lui sont connectés : les périphériques. Le problème posé est tel que l'utilisateur a tout intérêt à se renseigner précisément sur la façon dont son micro-ordinateur communique avec ses périphériques.

Deux types de périphériques sont prédisposés à poser des problèmes à leur possesseur à cause d'ennuis de transmission : il s'agit des modems et des imprimantes.

Pour que le micro-ordinateur et les périphériques « se comprennent », il faut configurer les ports; c'est pourquoi une option du menu des utilitaires-système vous propose de vous aider à le faire.

Si vous n'achetez que des matériels conçus pour l'Apple //c, vous êtes tranquille puisque toutes les configurations de ports préétablies vont fonctionner parfaitement.

Les ennuis commencent dès que l'on s'écarte du matériel dédié à l'Apple //c : l'imprimante parallèle de l'Apple //e ne communique pas naturellement avec l'Apple //c qui dispose de sorties série.

Avant de poursuivre, il est nécessaire de rappeler le vocabulaire particulièrement précis des transmissions.

La transmission parallèle est un mode de transmission dans lequel tous les bits d'un caractère sont transmis simultanément par des voies de communication physiquement distinctes.

C'est le cas des échanges à l'intérieur du micro-ordinateur entre le microprocesseur et les autres dispositifs; les voies de transmission s'appellent les bus.

Ce mode de transmission semble être le plus simple et le plus évident, mais certains inconvénients peuvent se révéler parfois gênants : par exemple, il y a au moins huit fils de cuivre en parallèle, ce qui peut être gênant sur de longues distances. On reproche aussi à ce mode de transmission d'être très sensible aux parasitages et donc de favoriser les pertes d'informations.

Ces ennuis n'existent pas avec le mode de transmission « série » : tous les bits d'un caractère sont envoyés successivement et par le même chemin.

Rappelons que les caractères disponibles sur l'Apple //c sont codés grâce au code ASCII, et qu'ainsi 7 bits suffisent pour un seul caractère. On ajoute en général un huitième bit qui indique la parité du signal envoyé, permettant ainsi de détecter une erreur dans la transmission d'un caractère si un bit est mal transmis.

Pour connecter vos périphériques, vous devez donc vous assurer que les sorties qu'ils exigent sont compatibles avec les interfaces proposées en sortie de l'Apple //c.

Une interface est un dispositif qui permet à deux systèmes distincts de communiquer. Il adapte les signaux suivant le mode de transmission. L'Apple //c a décidé pour vous. Il est équipé de deux interfaces série RS 232 C.

Les deux sorties correspondent au port 1 (configuré par défaut pour l'imprimante ImageWriter et au port 2 (configuré par défaut pour un modem 300 bauds).

Ces deux ports sont reconfigurables pour d'autres périphériques en transmission (imprimante) ou communication (modem) : voir toujours l'« utilitaires-système » correspondant. Le NIP (numéro d'identification de périphérique) caractérise tous les détails d'une transmission série.

En effet, deux modes de transmissions série sont possibles :

— *Le mode asynchrone* dans lequel les caractères sont transférés un par un, mais encadrés de bit de début (bit start) et de bit de fin (bit stop). Le récepteur est ainsi synchronisé et il guette les formats des bits qui arrivent pour trier les données des caractères de contrôle.

— *Le mode synchrone* pour lequel l'émetteur et le récepteur doivent être accordés sur le rythme de la transmission. Le transfert est cadencé par une « horloge » et le débit des bits des caractères successifs est constant. On se passe des bits start et stop mais à la moindre erreur, tout le message devient incompréhensible.

Le modèle asynchrone est le plus simple d'emploi mais il ralentit la transmission.

La vitesse de transmission est en micro-informatique le bit par seconde, appelé encore baud.

L'interface RS 232 C a été proposée par Electronic Industriès Association; le C signifie qu'elle en est à sa troisième révision.

Caractéristiques :

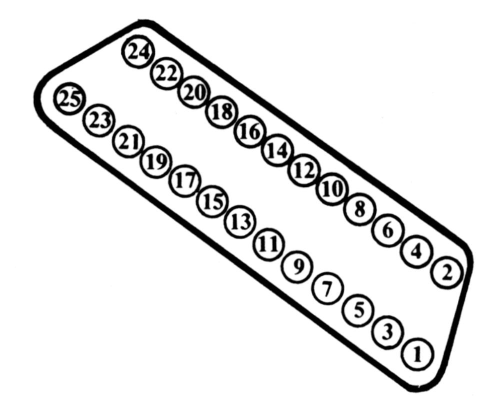
— elle fonctionne en mode asynchrone (bit start, 5 à 8 bits de données, bit stop);

— vitesse de transmission de 50 à 9 600 bauds;

— niveaux logiques : 5 à 25 V pour le 0, – 5 à – 25 V pour le 1;

— le connecteur est normalisé, vous en trouverez deux derrière votre Apple //c.

1 Terre de protection
2 Émission de données
3 Réception de données
4 Demande pour émettre
5 Prêt à émettre
6 Poste de données prêt
7 Terre de signalisation
8 Détecteur de porteuse
15 Horloge émission interne
17 Horloge réception
20 Terminal de données prêt
24 Horloge émission externe



Remarque : après quelques aménagements, une liaison série peut supporter des communications bidirectionnelles (dans les deux sens).

L'ensemble des manœuvres nécessaires aux échanges entre le micro-ordinateur et ses périphériques s'appelle un *protocole de transmission*. L'ensemble des signaux préparant et précisant la transmisson s'appelle le *handshaking* (poignée de main), illustrant le fait que pour échanger des données, il faut que l'émetteur et le récepteur soient prêts simultanément.

LES MONITEURS

La différence fondamentale entre votre téléviseur habituel et un moniteur vendu pour servir de périphérique est que le moniteur peut afficher des caractères plus petits sans que l'écriture devienne floue.

Un téléviseur est plutôt conçu pour montrer des images, alors qu'un moniteur est prévu pour afficher les caractères d'un texte. L'utilisation d'un traitement de texte réclame un moniteur alors que le téléviseur suffira pour les applications ludiques de votre Apple //c.

De manière générale, on ne conçoit pas d'utilisation professionnelle d'un micro-ordinateur s'il n'est pas équipé d'un moniteur.

L'Apple //c satisfait aux deux applications possibles de la micro-informatique. Pour son usage dans la version « familiale », il est doté d'un cordon Péritel permettant son raccordement à un téléviseur (couleur ou noir et blanc, les deux existent).

Pour une utilisation professionnelle, la gamme Apple dispose de plusieurs moniteurs construits spécialement pour le //c.

Caractéristiques d'un écran ou d'un moniteur

La dimension : elle est le plus souvent exprimée en « pouces » (unité anglo-saxonne, 1 pouce = 25,4 mm) et le chiffre donné correspond alors à la longueur de la diagonale de l'écran.

La définition : c'est la distance entre deux éléments affichés. Plus cette distance est faible, meilleure est la définition.

Remarque : la dimension influe sur la définition de l'image; en général, plus la dimension de l'écran est grande, moins la définition est bonne.

La précision : c'est-à-dire la différence entre ce qui est demandé et ce qui est affiché. Par exemple, lors de l'utilisation d'un crayon optique, c'est la distance qui sépare la position pointée de la position enregistrée.

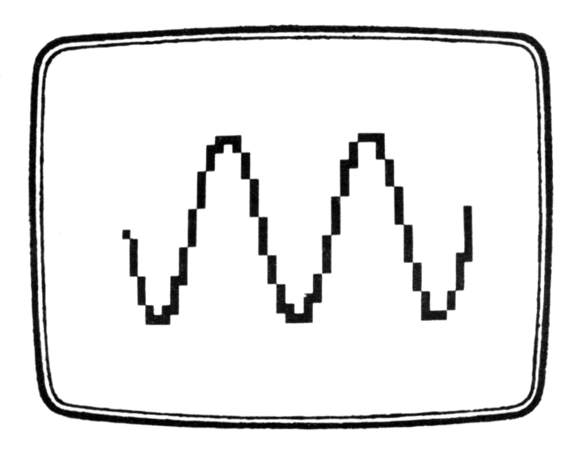
Un écran possède d'autres caractéristiques qui échappent à l'utilisateur. Une pourtant est particulièrement importante : il s'agit de la place mémoire que l'écran occupe dans la mémoire centrale de l'ordinateur.

Certaines zones de la mémoire correspondent à l'écran et le nombre de caractères affichés donne une idée de la mémoire d'écran.

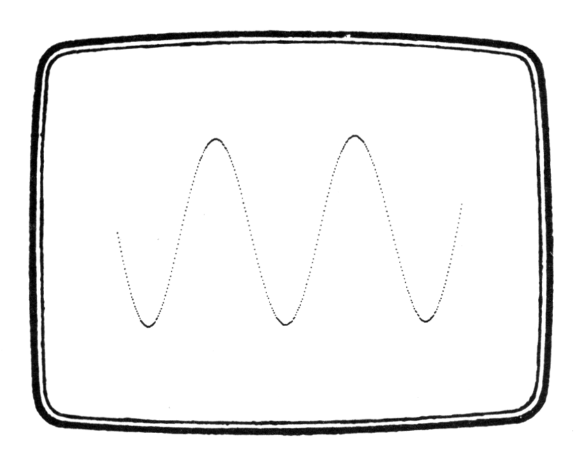
Pour avoir une indication sur l'organisation de votre Apple //c, les instructions BASIC comme PEEK et POKE vous permettront d'explorer la mémoire d'écran. A titre indicatif, la « page texte » se trouve entre les adresses 1Ø24 et 2Ø48.

Lorsque le micro-ordinateur affiche du graphisme haute résolution, il doit adresser chaque pixel de l'écran et non plus chaque caractère. En conséquence, le nombre d'adresses est multiplié par 64 si chaque caractère est défini par un pavé de 8 × 8 pixels.

Les pages graphiques haute résolution occupent donc beaucoup de place mémoire, ce qui est parfois pénalisant dans l'écriture d'un programme. Sur l'Apple //e, on demandait souvent aux utilisateurs du graphique haute résolution de se procurer des cartes d'extension mémoire.



Basse résolution



Haute résolution

La technique employée par les moniteurs et les téléviseurs repose sur un même principe : un faisceau d'électrons est dévié et guidé par un champ magnétique ; l'impact de ce faisceau sur un matériau luminescent produit un point lumineux : le pixel. La lumière ainsi obtenue s'efface graduellement et demande un balayage régulier pour que le dessin reste visible à l'écran. En général, un écran est balayé 25 fois par seconde.

La différence de technique entre le téléviseur et le moniteur vidéo intervient lors des transmissions des données à visualiser. Le principe TV consiste à moduler les informations à transmettre, l'appareil de visualisation étant muni d'un démodulateur et d'un filtre. Un moniteur, au contraire, transmet directement les données.

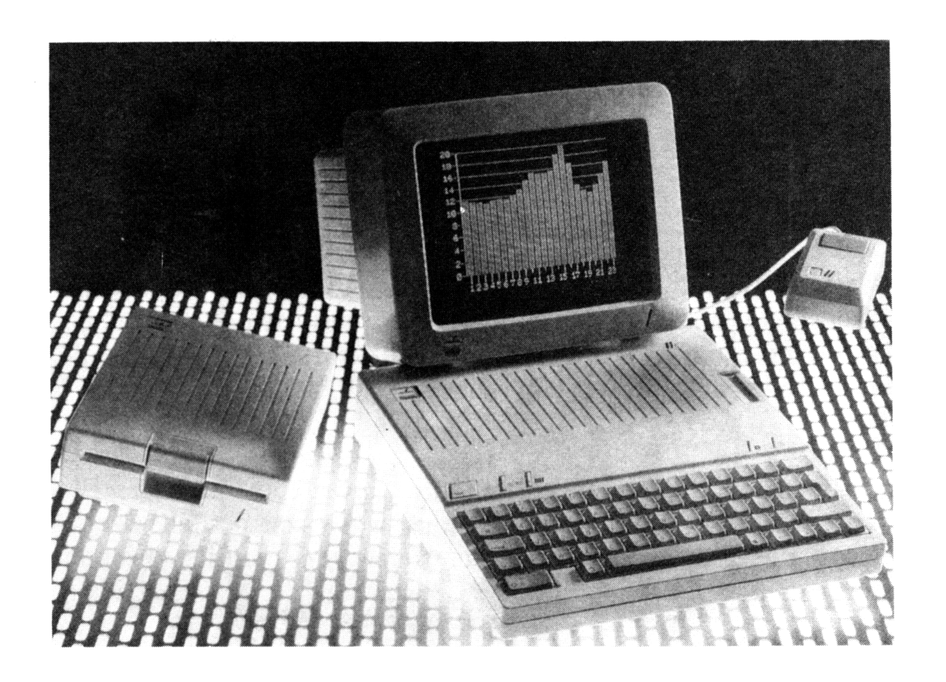
Conséquences : la résolution et la clarté des moniteurs sont bien meilleures que celles d'un téléviseur. De plus, à l'inverse des téléviseurs, les moniteurs sont insensibles aux interférences et parasites dus à la transmission.

La prise Péritel permet de rendre compatibles les signaux issus du micro-ordinateur (donc non modulés) avec le dispositif du téléviseur qui s'attend à recevoir des signaux modulés.

Les moniteurs étant des appareils professionnels, les qualités ergonomiques sont mises en avant : brillance et contraste lumineux, antireflet, luminosité, réglage, etc.

Le moniteur de l'Apple //c — conçu spécialement pour lui — est monochrome (il affiche en vert sur fond noir) et accepte le graphique haute résolution. Il est petit par la taille (la diagonale fait 9 pouces) et permet d'afficher 80 colonnes et 24 lignes (ce sont en fait les performances de l'Apple //c qui commandent ces deux derniers paramètres).

On peut lui adjoindre un petit support qui permettra une orientation et une position optimales devant l'utilisateur.



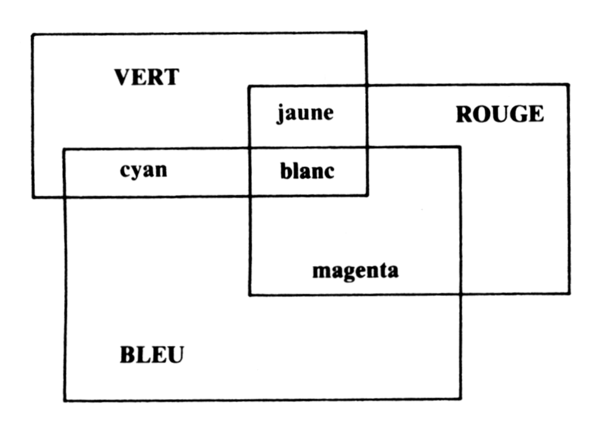
Le moniteur de l'Apple //c se connecte directement dans un port prévu à cet effet à l'arrière du micro-ordinateur.

On considère que la couleur d'affichage la moins fatigante pour les yeux est l'ambre sur fond noir dans le cas d'un moniteur mono chrome. Les moniteurs couleur proposent évidemment des avantages considérables sur les moniteurs monochromes : meilleure lecture immédiate, fatigue réduite...

Les écrans couleur

Les couleurs sont obtenues par un triple faisceau d'électrons, Rouge, Vert, Bleu d'où le nom du procédé : RVB.

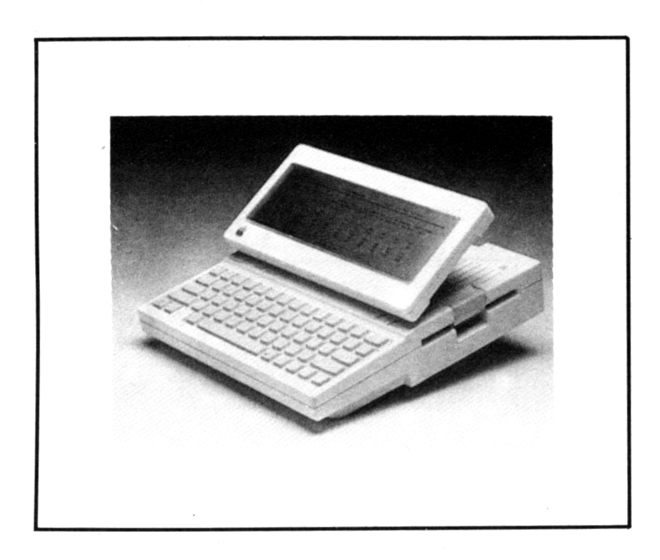
Les superpositions permettent d'obtenir quatre autres couleurs mais on observera que la combinaison des couleurs n'est pas la même que sur les rosaces traditionnelles montrant les couleurs primaires et les couleurs secondaires.



A l'affichage, l'œil perçoit la somme des rayons émis. Un écran couleur est recouvert d'une série de couches de luminescence différente. En pénétrant plus ou moins dans ces couches, le faisceau d'électrons provoque la visualisation des différentes couleurs.

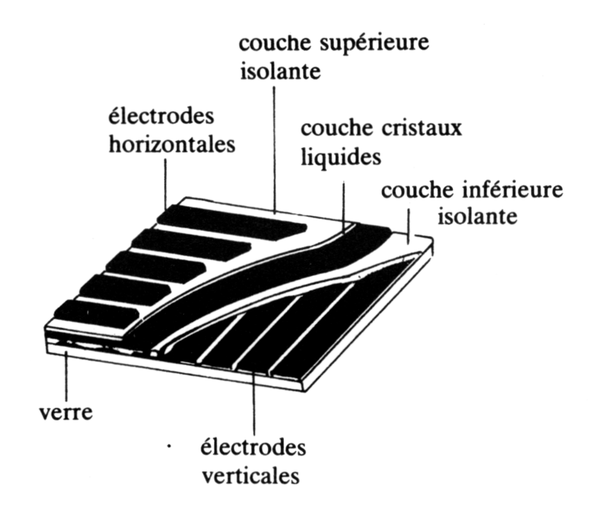
L'écran plat

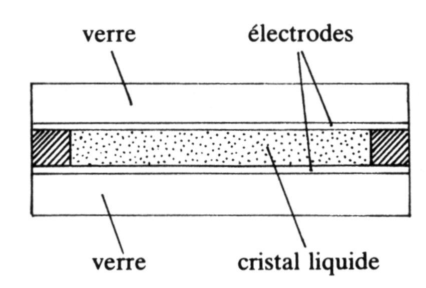
L'Apple //c est un micro-ordinateur portable et il est normal que la firme Apple ait cherché à développer un écran plat à cristaux liquides pour rendre le //c véritablement autonome.



Les cristaux liquides sont constitués de molécules en forme de bâtonnets. La position de ces bâtonnets les uns par rapport aux autres détermine leur propriétés optiques. Les cristaux liquides n'émettent pas de lumière ; ils se contentent de modifier les propriétés des faisceaux lumineux qui les traversent.

On ordonne les molécules composant les cristaux à l'aide d'un champ électrique et chaque bâtonnet est commandé par une électrode.





Le principal intérêt d'un écran à cristaux liquides est sa très faible consommation énergétique.

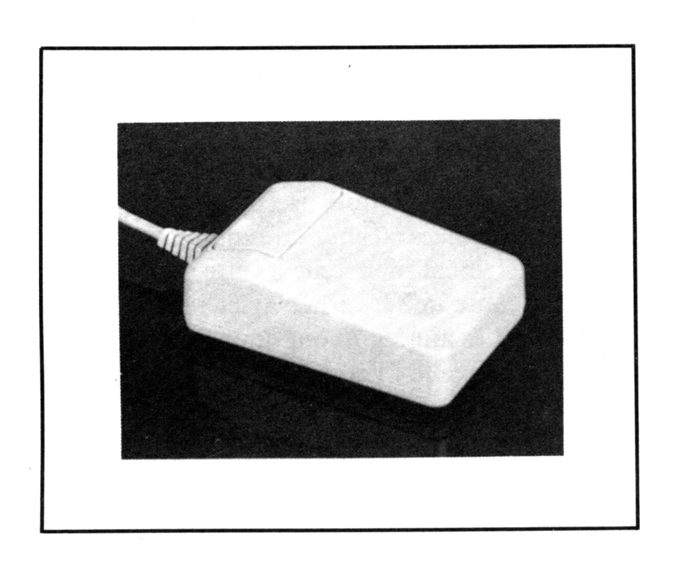
Les progrès de la technologie des microprocesseurs avaient déjà réussi à rendre les micro-ordinateurs très peu gourmands en électricité mais l'arrivée des écrans à cristaux liquides permet vraiment de rendre un micro-ordinateur complètement portable.

L'écran plat à cristaux liquides de l'Apple //c se caractérise donc par son faible poids et un encombrement réduit. Il est construit pour être fixé sur l'Apple //c, derrière le clavier. L'inclinaison de l'écran est très importante pour permettre une lecture agréable.

Les caractéristiques d'affichage de cet écran à cristaux liquides suivent celle de la machine, c'est-à-dire 24 lignes de 80 caractères.

LA SOURIS

Une souris se présente comme un petit bloc que l'on déplace sur un plan de travail et qui commande les déplacements du curseur à l'écran. L'utilisateur remarquera immédiatement la simplicité des manipulations à effectuer et la vitesse atteinte lors des mouvements du curseur. La souris dispose aussi d'une touche de validation : quand un menu est affiché à l'écran, on sélectionne une option en déplaçant le curseur sur l'option choisie puis en "cliquant" sur la touche de validation. Le micro-ordinateur comprend alors l'ordre reçu.



La souris apporte un avantage évident à l'utilisateur de progiciels. Pour les traitements de texte, les tableurs, les systèmes de gestion de base de données... on a déjà fait remarquer la meilleure convivialité d'un progiciel dirigé avec la souris par rapport à un progiciel commandé entièrement au clavier. L'ensemble des programmes actuellement disponibles sur l'Apple //c montre clairement une orientation délibérée vers les "logiciels avec souris".

Des solutions différentes ont été proposées sur d'autres micro-ordinateurs pour s'affranchir des frappes au clavier. Le crayon optique et l'écran tactile reposent sur un même principe : l'utilisateur doit toucher l'écran avec le crayon ou son doigt pour désigner l'option désirée. Ceci oblige l'opérateur à travailler près de l'écran et à lever souvent le bras ; en revanche, les mouvements sont plus rapides.

Les joysticks ne sont utilisés que pour commander les logiciels de jeux.

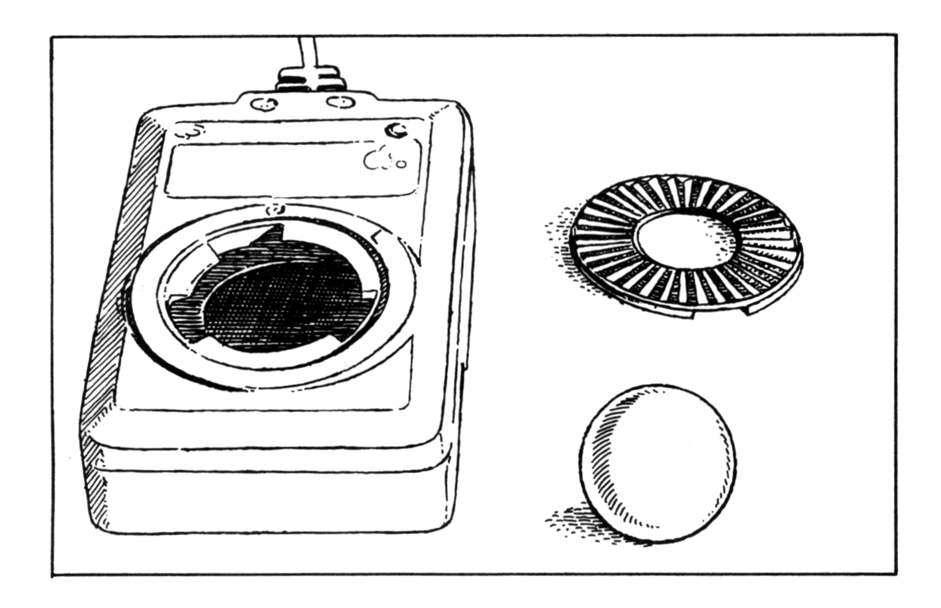
Petite histoire de la souris

Ce périphérique au nom charmant a marqué l'histoire de la micro-informatique.

On pourrait raconter "l'irrésistible ascension de la petite bête". La souris est née dans les années 60 chez une équipe de chercheurs américains. Passée complètement inaperçue à cette époque, nul ne lui prédisait un avenir de conte de fées : le clavier régnait en maître sur la micro-informatique. Les années 70 ont donné ses lettres de noblesse à une science presque nouvelle : l'ergonomie. La micro-informatique a été l'objet de soins particuliers et les ergonomes se sont préoccupés des conditions de travail devant un écran. Apple a démontré tous les avantages que l'on pouvait tirer du "petit animal" avec Lisa.

Aujourd'hui, le développement des logiciels bénéficiant de la souris montrent à quel point elle est devenue un périphérique privilégié et un outil simplificateur.

Principe de la souris de l'Apple //c



Sous le boîtier apparent, la plage ? Non, une boule ! Le nombre de tours qu'elle effectue commande le déplacement du curseur et toutes les directions sont possibles. Pour se repérer, la souris se sert de coordonnées relatives (comptées en tours de boule), ce qui laisse une grande latitude dans l'utilisation de son espace de travail : on peut la soulever et reprendre un déplacement sans problème. La souris se connecte à l'Apple //c au port marqué d'une double icône : celui de la souris et celui des manettes de jeux. La fiche de connexion est une fiche à 9 broches et le port est à l'arrière du boîtier de l'Apple //c.

La souris de l'Apple //c s'appelle APPLE MOUSE tout simplement. Elle est livrée avec un progiciel de dessin MOUSEPAINT. Ce logiciel permet de faire des diagrammes, des cartes, du lettrage et bien sûr des dessins. Le menu proposé se commande à l'aide de la souris et les fonctions permises sont symbolisées par un crayon, une gomme, des plumes, une corbeille à papiers, etc. Les résultats sont surprenants.

Remarque : la même souris équipe Lisa et le Macintosh ; vous avez ainsi la puissance des "grands" au service de votre Apple //c !

LES IMPRIMANTES

La première remarque concernera le protocole de transmission ; d'après le chapitre précédent, l'Apple //c ne tolère — en première approche — que les périphériques disposant d'une sortie série. Or beaucoup d'imprimantes fonctionnent sous le mode parallèle (n'oublions pas que l'interface parallèle Centronics a été développée par un constructeur d'imprimantes). L'Apple //e avait d'ailleurs une sortie parallèle pour son imprimante. Il faudra donc prêter attention aux connexions même à l'intérieur de la gamme Apple.

Une imprimante est un périphérique rapidement indispensable aux utilisateurs de micro-ordinateurs. En effet, il n'est pas longtemps concevable de ne visualiser son travail que sur un écran. Les sorties de listes sur papier sont nécessaires.

On caractérise une imprimante par :

- le jeu de caractères disponibles et imprimables,
- le nombre de caractères par ligne (en cpl),
- le type de papier et le mode d'entraînement,
- le sens d'impression, c'est-à-dire unidirectionnel : impression de gauche à droite, bidirectionnel : impression à l'aller et au retour du chariot, bidirectionnel optimisé : bidirectionnel avec optimisation du trajet du chariot,
- la capacité de la mémoire tampon,
- le mode d'impression.

LES DIFFÉRENTS MODES D'IMPRESSION

Les imprimantes matricielles

Elles utilisent une matrice de points stockée en PROM pour définir un caractère. Le nombre de points utilisés conditionne la qualité, la précision et les possibilités de reconstitution d'un caractère : dessiner un ç ou un ê demande un nombre important de points.

Les imprimantes matricielles s'opposent aux imprimantes à marguerite (ou à boule) pour lesquelles les caractères sont formés sur la marguerite.

Les imprimantes à impact

Les imprimantes à impact utilisent la frappe de la tête d'écriture (qu'elle soit matricielle ou non) contre un ruban encré, exactement comme pour une machine à écrire. L'encre se dépose sur le papier placé derrière le ruban.

Autres imprimantes

A l'opposé de ce système on trouve :

— les imprimantes thermiques qui utilisent un papier spécial que l'on brûle à l'aide d'un composant thermique ;

— les imprimantes à jet d'encre pour lesquelles l'impression s'effectue par projection de fines gouttelettes d'encre. Elles portent chacune une petite charge électrique, et un champ magnétique les dévie et les guide vers la position désirée ; le séchage est immédiat.

Ces diverses techniques reposent sur des choix différents. Les imprimantes matricielles sont en général assez rapides et relativement peu coûteuses. Elles permettent de faire du graphisme haute résolution. Elles ont cependant un gros défaut : il faut que la matrice ait vraiment beaucoup de points pour que l'aspect "petits points noirs" ne soit pas évident.

Les milieux professionnels préfèrent alors les machines dont les caractères sont déjà formés (marguerite, boule), pour une impression ayant la "qualité courrier".

Les imprimantes à marguerite sont en général assez lentes. Il faut que la marguerite tourne sur elle-même entre chaque frappe de caractère : c'est donc le problème technique des mouvements mécaniques qui ralentit la vitesse de frappe.

A titre indicatif, les imprimantes à marguerite les moins rapides ont une vitesse de 15 à 20 caractères par seconde et les plus rapides ne dépassent guère 60 caractères par seconde. Le prix assez élevé des imprimantes à marguerite les réserve en général aux usages professionnels nécessitant un qualité d'impression similaire à celle fournie par une machine à écrire (à marguerite !).

Pour essayer de combler le handicap de la "qualité courrier", les constructeurs d'imprimantes à aiguilles ont d'abord augmenté le nombre d'aiguilles puis proposé des améliorations comme le double passage : chaque caractère est imprimé deux fois avec un léger décalage de la tête d'impression entre les deux passages, ce qui permet de combler les trous.

La grande qualité des imprimantes matricielles est la vitesse d'impression très appréciable. Il n'est pas rare de trouver des vitesses de l'ordre de 200 caractères par seconde quand elles utilisent les options comme la frappe bidirectionnelle : quand la tête d'écriture est parvenue en bout de ligne, il est peu rentable de la faire revenir systématiquement à la marge gauche. Certaines imprimantes peuvent imprimer la ligne suivante en profitant du retour vers la marge gauche.

Les autres techniques répondent à des besoins plus spécifiques.

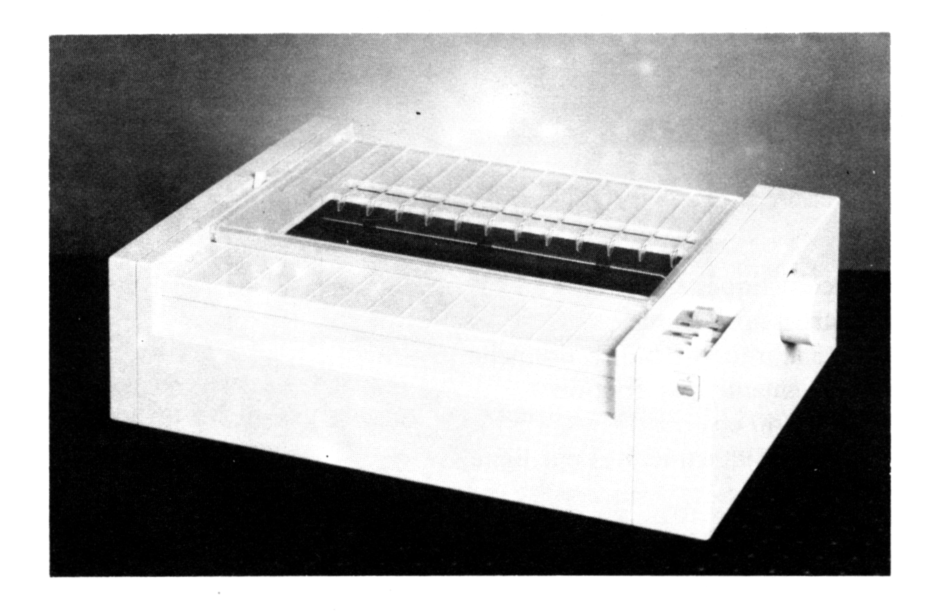
Les imprimantes thermiques sont des imprimantes matricielles pour lesquelles chaque point de la matrice peut être noirci par un composant thermique placé sur la tête d'écriture. Elles utilisent un papier spécial (thermosensible) et leur grande qualité vient du silence du procédé d'impression. Malheureusement, la conservation des documents sur papier thermosensible est problématique car les points brûlés ont tendance à se ternir sous l'action du soleil et de l'air. Ce type d'imprimante est souvent connecté aux petits ordinateurs portables.

Remarque : l'imprimante thermique classique connectable aux micro-ordinateurs Apple s'appelle Silentype et est construite par Apple lui-même. Cependant, l'interface parallèle interdit pour l'instant sa connexion à l'Apple //c. On peut penser qu'une adaptation sera bientôt disponible.

On trouve aussi un procédé d'impression sur papier métallisé. Elles ont les mêmes objectifs avoués que les imprimantes thermiques (silence et prix). Un point noir de la matrice est obtenu grâce à la création d'un arc électrique entre une électrode de la tête d'impression et le papier.

LES IMPRIMANTES DE L'APPLE //c

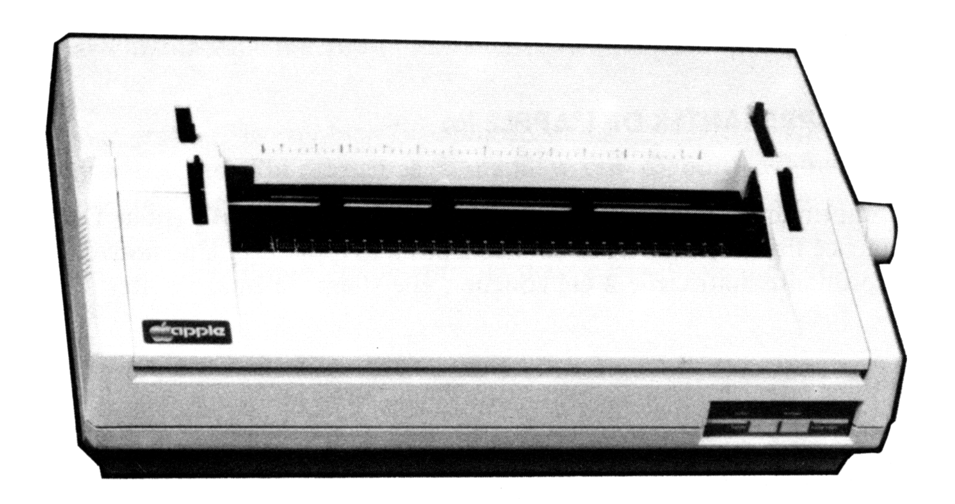
L'imprimante classique de l'Apple //e s'appelait la DMT (pour Dot Matrice Printer). Elle ne se connecte pas à l'Apple //c. Une nouvelle imprimante matricielle a été conçue : elle s'appelle ImageWriter.



Caractéristiques :
Constructeur : Apple
Type : matricielle bidirectionnelle graphique
Entraînement : mixte
Vitesse : 180 cps
Largeur : 150 caractères par ligne

Notons aussi que le prix de cette imprimante est très bas pour les performances annoncées.

Apple a développé une imprimante à marguerite pour le marché professionnel demandant la qualité courrier.



Caractéristiques :
Constructeur : Apple
Type : marguerite bidirectionnelle
Entraînement : par friction
Vitesse : 40 cps
Largeur : 198 caractères par ligne

Il faut signaler que le prix est très élevé et réserve ce matériel à des usages professionnels.

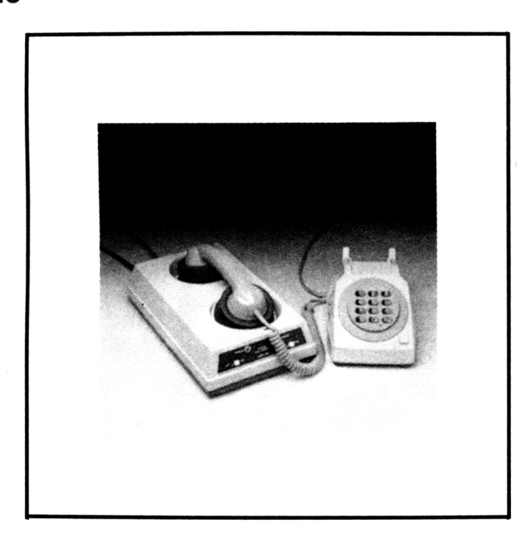
Signalons aussi l'existence d'une imprimante à jet d'encre, la SIEMENS PT88/PT89 dont les caractéristiques techniques sont :
Vitesse : 80 cps ou 150 cps selon le modèle
Largeur : 136 caractères par ligne.

La liste des imprimantes connectables à l'Apple //c est longue. Nous avons choisi ici quelques matériels classiques pour vous signaler que tous les types d'imprimantes sont disponibles pour le //c.

Il est évident que toutes les imprimantes ont des sorties série RS 232 C ; mais elles peuvent également disposer d'une sortie parallèle pour être connectées à d'autres micro-ordinateurs.

Le port de sortie pour imprimante porte le numéro 1. Il est placé derrière l'Apple //c.

LES MODEMS



Un modem est un périphérique servant à connecter l'ordinateur au réseau téléphonique.

Les informations stockées dans un micro-ordinateur sont codées. Il est possible de faire voyager ces informations sur des réseaux de communication, à condition de respecter les caractéristiques de ce réseau. Les bits à transmettre doivent donc subir une préparation qui s'appelle une modulation. Le dispositif récepteur retraduira les signaux en caractères alphanumériques.

Un modem permet à l'émission de moduler les signaux binaires provenant du micro-ordinateur et à la réception, il démodule le signal analogique et reconstitue les 1 et les 0 du code binaire.

Remarque : le nom de "modem" est une contraction de modulateur - démodulateur.

La modulation utilise un signal sinusoïdal. Les caractéristiques de ce signal sont l'amplitude, la fréquence et la phase. Si le signal à transmettre est une suite de 0 et de 1, on modifiera un ou plusieurs paramètres de la porteuse (la porteuse = le signal sinusoïdal) en fonction de la présence d'un 1 ou d'un 0.

Remarque : on ne peut pas transmettre correctement des signaux sur de longues distances sans les moduler. La modulation permet de diminuer les "bruits" (parasites). Elle réduit également les problèmes liés à l'amortissement et à la distorsion des signaux.

Différents modems sont disponibles actuellement :

• les modems acoustiques pour lesquels on dépose le combiné du téléphone dans un réceptacle adéquat. Le modem émet alors des signaux "audibles" qui sont transmis suivant le même principe que pour la voix humaine.

• les modems (non acoustiques) qui disposent d'une prise pouvant se substituer à une prise de téléphone. On rebranche le téléphone au modem. Les signaux transmis par le réseau téléphonique sont des signaux électriques et c'est sous cette forme que le modem les émet. On évite donc une retraduction acoustique.

La différence entre ces deux techniques rend la seconde plus sûre et plus performante. Elle la rend également plus chère.
Les possesseurs de l'Apple //c, pour qui l'aspect portable de leur machine fut un critère décisif d'achat, s'intéresseront particulièrement au modem pour la communication à l'intérieur de l'entreprise. Le //c devient ainsi un terminal capable de communiquer avec une base de données distante.

Deux modems sont actuellement disponibles pour l'Apple //c. Ils se distinguent principalement par la vitesse de transmission : le premier permettant une vitesse de 300 bauds, le second de 1 800 bauds.

L'interface RS 232 C a été développée à l'origine pour les modems. Vous ne serez donc pas surpris d'apprendre que les modems d'Apple l'utilisent systématiquement.

Le port de connexion des modems se trouve à l'arrière de l'Apple //c et porte le numéro 2.

LE LECTEUR DE DISQUETTES

Dès qu'une application dépasse un certain niveau de complexité, la configuration requise pour la réaliser peut exiger deux lecteurs de disquettes. C'est le cas de certains logiciels — on a vu Thinktank par exemple —, c'est aussi le cas du langage PASCAL.

Avoir deux disquettes en liaison avec le micro-ordinateur ne double pas simplement le nombre d'informations stockées dans la mémoire de masse, on peut aussi les ordonner de manière particulière. C'est le cas de PASCAL ou Thinktank qui séparent les programmes et les données sur chaque disquette. On a également vu que le système PRODOS permettait de considérer les deux unités de disquettes comme une seule sortie logique et pouvait ainsi alterner la recherche sur chacune des deux disquettes.

La connexion d'un second lecteur de disquettes altère le caractère portable de l'Apple //c. Elle permet en contrepartie d'augmenter sa puissance et surtout de faciliter les manipulations de l'utilisateur.

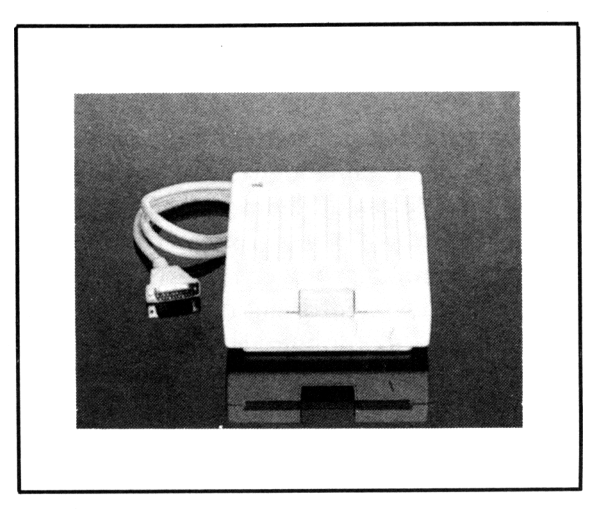
Exemple : on vous conseille vivement de faire des copies de la disquette utilitaires-système. En n'utilisant que le lecteur de disquettes intégré vous avez dû constater que le nombre de manipulations nécessaires était assez important : ''introduire le disque source'' et ''introduire le disque destination'' sont deux commandes à répéter une dizaine de fois. Le second lecteur de disquettes vous dispense de ces manœuvres répétitives et dangereuses puisqu'une interversion des deux disquettes peut conduire à la catastrophe.

Attention : tous les lecteurs de disquettes disponibles chez Apple ne se connectent pas au modèle //c. En particulier, le mono-disk du //e.

Un lecteur de disquettes a spécialement été construit pour se connecter à l'Apple //c.

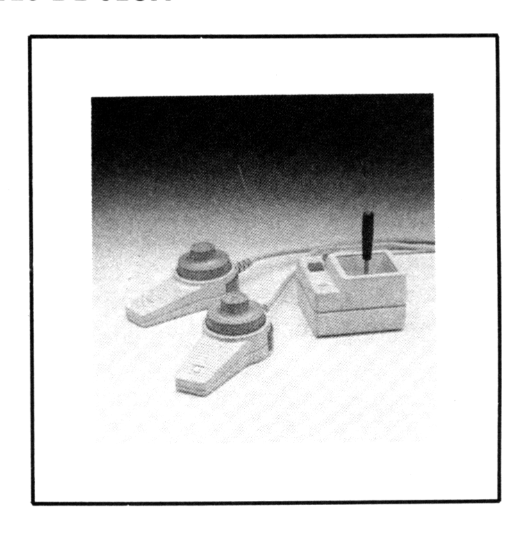
Il est la réplique exacte du lecteur de disquettes intégré ; seul change le capot qui l'entoure.

Ses caractéristiques sont celles des autres lecteurs de disquettes des Apple // : elles ne permettent qu'un stockage de 140 Ko sur une face de disquette.



Le port de sortie pour le second lecteur de disquettes se trouve à l'arrière de l'appareil. On remarquera que les prises sont toutes différenciées et qu'il est impossible de confondre le port du lecteur avec le port du téléviseur.

LES MANETTES DE JEUX



La réputation des jeux d'Apple n'est plus à faire : il existe des dizaines de logiciels sur le marché tels "Donjons et Dragons" qui a connu le plus vif succès, les jeux de réflexe, d'adresse. Mais ces derniers utilisent des périphériques particuliers : les manettes de jeux.

Trois sortes de manettes existent :
— les manettes à potentiomètre souvent désignées par leur appellation anglaise : *paddle*,
— les manettes à manche à balai ou *joystick*,
— les manettes à boule dont le fonctionnement est celui d'une souris que l'on aurait retournée.

Les manettes de jeux utilisent le même principe que la souris pour déplacer un indicateur sur l'écran. Elles se connectent d'ailleurs au même port derrière le capot de l'Apple //c.

La précision du déplacement n'est pas la première qualité demandée aux manettes de jeux. On est surtout sensible à la résistance qu'elles apportent aux débordements d'enthousiasme ou de dépit de leur utilisateur. Tous les réflexes sauvages sont à éviter et aucune garantie n'est donnée en cas de crise de nerfs ! Bonne chance.

DISQUE DUR

Il est aujourd'hui disponible en France, un disque dur de 3,5 pouces de diamètre et d'une capacité de stockage de 10 Méga-octets commercialisé par la société Symbiotic Computers Systems.

Ce produit est compatible avec toute la gamme Apple et supporte les systèmes d'exploitation DOS 3.3, PASCAL, et PRODOS. Il se charge directement sur le port du deuxième lecteur du //c.

CONCLUSION

Au dire de son constructeur, l'Apple //c est une machine « ambigüe, polyvalente et multimode ».

Le portable d'Apple a donc autant l'ambition de se laisser transporter que de transporter son utilisateur dans la direction qui lui conviendra. Côté logiciel, on appréciera de disposer de l'immense bibliothèque de l'Apple //e qui couvre la plupart des besoins, depuis les jeux jusqu'aux logiciels professionnels. Le //c bénéficie en cette matière d'un avantage considérable : quel que soit l'avenir commercial de cette machine, elle est assurée d'un environnement logiciel très confortable.

Mais les principaux attraits du //c sont à chercher ailleurs, du côté des périphériques. Quoique la gamme des imprimantes, modems et tablettes graphiques compatibles Apple reste à la disposition du //c, ce sont surtout l'écran plat et la souris qui doivent retenir l'attention de l'utilisateur et du futur utilisateur.

On peut être étonné que la vague de miniaturisation n'ait pas emporté les périphériques de visualisation avec autant d'ardeur qu'elle l'a fait en ce qui concerne les périphériques de stockage et l'unité centrale. En surface, un écran se doit de conserver une taille raisonnable compte tenu des contraintes organiques de l'œil, mais en profondeur, la technique des cristaux liquides permet enfin de s'affranchir de l'encombrement (et de la fragilité) des classiques tubes à vide.

En ce qui concerne les écrans monochromes, l'écran plat prouve que la technique est au point. En réduisant les dimensions et la consommation électrique, elle donne à l'Apple //c l'opportunité de mériter pleinement la qualité de portable. Il reste encore à attendre l'écran plat couleur au sujet duquel il n'est question que de rêver...

Enfin, la souris... A elle seule, elle s'offre le luxe d'étonner les habitués de l'informatique et de conquérir les néophytes. Déjà, des logiciels professionnels, spécialement conçus pour le //c équipé d'une souris, démontrent que les concepteurs de logiciels voient en elle le partenaire presque obligé du //c. Il s'agit d'un traitement de texte (EPISTOLE), d'un intégré (JANE) et d'un tableur (VERSION-CALC). A ce jour, nous attendons encore un logiciel de jeu dans lequel une meute de chats s'en prendrait à une malheureuse... souris.

Achevé d'imprimer le 15 janvier 1985
sur les presses de l'imprimerie Carlo Descamps
à Condé-sur-l'Escaut 59163-France

Dépôt légal : janvier 1985 — N° d'imprimeur : 3683

Imprimé en France

APPLE IIc, VOTRE MICRO-ORDINATEUR

Christophe Doë et Éric Duceau

La famille Pomme s'agrandit...
L'Apple II C portable a de quoi séduire les utilisateurs les plus exigeants : un faible encombrement et des possibilités de traitement exceptionnelles pour un si petit gabarit.
Ce livre vous fera découvrir les caractéristiques techniques, les logiciels et les périphériques qui font de l'Apple II C un outil de travail et de loisir hors du commun.





**

291315

ISBN - 2 - 7124 - 1517 - 5

APPLE IIc, VOTRE MICRO-ORDINATEUR

cedic
nathan